Project Report

PA-229-12
(RSP)

C. R. Berndtson
R. H. French
D. E. Nessman

Data Reduction Program Documentation ALTCEP

(Effective: July 1971)

8 July 1971

Prepared for the Advanced Research Projects Agency,
the Department of the Army, and the Department of the Air Force
under Electronic Systems Division Contract F19628-70-C-0230 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Lexington, Massachusetts

Distribution limited to U.S. Government agencies only; test and evaluation; 9 July 1971. Other requests for this document must be referred to ESD-TR-2.

90

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

6

DATA REDUCTION PROGRAM DOCUMENTATION ALTCEP 9 (EFFECTIVE: JULY 1971),

10

Charles

C. R. BERNDTSON

[redacted]

R. H. FRENCH
D. E. NESSMAN
Philco-Ford Corporation
Editors

14 PA-229-12

2ke

9

PROJECT REPORT ~~PA-229-12 (RSP)~~

11 8 JULY 1971

12 80p.

15 F19628-70-C-0230, ARPA Order-600

Distribution limited to U.S. Government agencies only; test and evaluation; 9 July 1971. Other requests for this document must be referred to ESD-TR-2.

L.G. Hanscom Field
Bedford, Mass 01731

LEXINGTON MASSACHUSETTS

207 650

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. This work was sponsored by the Advanced Research Projects Agency of the Department of Defense (ARPA Order 600), the Department of the Army, and the Department of the Air Force under Air Force Contract F19628-70-C-0230.

FOREWORD

This is the twelfth report in the Data Reduction Program Documentation series. It is dated according to the date of completion of the documentation. No implication is made that this program will not subsequently be modified, amended, or superseded; on the contrary, the history of radar data processing is one of continuous evolution of techniques, and it is unrealistic to assume that steady-state has been reached.

The preparation of reports in this series is under the Editorship of Charles R. Berndtson of Lincoln, and of D. Nessman and R. French of Philco-Ford Corporation. Inquiries, suggestions, corrections, criticisms, and requests for additional copies should be directed to C. R. Berndtson.

The principal contributor to this report was L. C. Zemke (Philco-Ford). Due to the intricate, evolutionary manner in which the programs came into being, the editors regret that it is in general impossible to give due credit to all -- mathematicians or radar analysts or programmers -- who contributed to the definition and writing of the programs.

Alan A. Grometstein

Alan A. Grometstein

CONTENTS

CONTENTS (cont'd)

CONTENTS (cont'd)

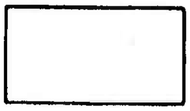
COMMON SYMBOLS AND ABBREVIATIONS

(The units given for certain quantities are the units commonly used for those quantities, unless otherwise noted.)

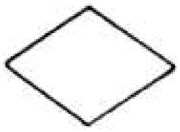
ADT	ALCOR Data Tape
ALCOR	ARPA-Lincoln C-band Observables Radar
ALTAIR	ARPA Long-Range Tracking and Instrumentation Radar
Alt	Altitude (km)
APS	Average Pulse Shape
ARS	ALTAIR Recording System
ARTP	ALTAIR Real Time Program
ATC	Angle Track Console
Avg	Average, Averaging
Az	Azimuth (deg)
c	Speed of Light
CADJ	Adjusted Calibration Constant (db)
C-band	ALCOR frequency, 5664 MHz (NB) and 5667 MHz (WB)
DBLT	Wide Band Pulse Doublet
DCO	Designations and Communications Operator
El	Elevation (deg)
EOF	End of File
GMT	Greenwich Mean Time
h	Hours
Hz	Hertz
IF	Intermediate Frequency
in	Inches
IRV	Inter-Range Vector
LC	Left Circular Polarization
lsb	Least Significant Bit
min	Minutes
NB	Narrow Band
NRTPOD	Non-real Time Precision Orbit Determination Program
POD	Project PRESS Operation and Data Summary Report
Phase	Presented in deg
PRF	Pulse Repetition Frequency (pps)
PRI	Pulse Repetition Interval (s)
pps	Pulses per second
pts	Points

R	Range (km)
$\dot{R}$	Range Rate (km/s)
rad	Radians
RC	Right Circular Polarization
RCS	Radar Cross Section (dbsm)
RF	Radio Frequency
RGC	Receiver Gain Control
RTC	Range Track Console
s	Seconds
SD_w	Standard Deviation of Wake Velocity
SDBLT	Wide Band Slaved Pulse Doublet
S/N	Signal-to-noise Ratio
T	Time
TAL	Time After Launch (s)
TGC	Transmitter Gain Control
Tr	Traverse Angle (deg)
UHF	ALTAIR Frequency; 415 MHz
V	Velocity
V_d	Doppler Velocity
V_w	Mean Wake Velocity
VHF	ALTAIR Frequency; 155.5 MHz
WB	Wide Band
WBS	Wide Band Slaved
WTR	Western Test Range
θ	Total Off-axis Angle (deg)
λ	Wavelength
*	Denotes Multiplication

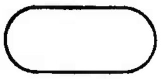
FLOW DIAGRAM SYMBOLS



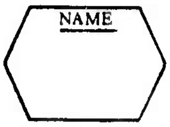
PROCESS, ANNOTATION



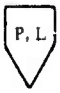
DECISION



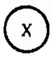
TERMINATOR



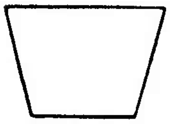
SUBROUTINE: where NAME is the entry call into the subroutine



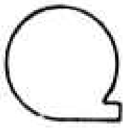
CONNECTOR: where P specifies a page in the flow diagram, and L designates a statement number in the program listing or a reference point in the flow diagram



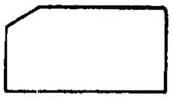
CONNECTOR: where X implies a continuation of the diagram to the next page



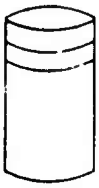
INPUT/OUTPUT OPERATION



MAGNETIC TAPE



PUNCHED CARD



DISK

ALTCEP

I. PURPOSE AND UTILIZATION

A. Source of Data

ALTAIR[1]

B. Data Input

ALTAIR catalog tape

C. Description

ALTCEP is designed to summarize ALTAIR's performance by examining a catalog tape. The program is normally run every 0.1 s. A test to tentatively identify which ALTAIR recording system ARS target is the target in track by ALTAIR is also performed.

D. Output

1. Listing of metric and radar status data.
2. Listing giving results of ARS target test.
3. RCS plots vs TAL of target in ALTAIR tracking gate.

continue on page 2

from page 1

II. DESCRIPTION

ALTCEP examines an ALTAIR catalog tape for radar status, performance, and target ~~RCS~~ radar cross section. It produces a listing (at a nominal interval of 0.1 s) of status and RCS data and plots of RCS vs TAL (VHF and UHF LC and RC).

The computation of metric data depends on the ARTP version[#] used at Kwajalein. Since 12 March 1971, R, Az, and El are corrected for known bias errors prior to recording on the catalog tape. Before 12 March 1971, R, Az, and El were corrected for bias errors by the ALTCEP program.

Vital format tables[##] are tested through a call to CHEAD, and if any are missing the program terminates.

The following paragraphs explain each item listed by ALTCEP.

A. GMT

ALTCEP uses GMT from two locations on the catalog tape.

The GMT found in FMRDRM, Items 1 and 2, is used with the metric listing. GMT found in FMCATF, Items 3 and 4, is used with the ARS target listing.

GMT (FMCATF) is compared to GMT (FMRDRM), and if they differ by > 1 s, the record is skipped and an error message printed.

B. TAL

TAL is GMT - lift-off time. Lift-off time may be input on data cards or read from the tape. If lift-off time is not input on data cards, ALTCEP examines Calibration Record FMGLOT, Item 1, to determine if the lift-off time in FMGLOT, Items 2 and 3, is valid. If lift-off time is valid, it is used. If it is not valid, it is set equal to zero, and TAL equals total GMT s.

[#] Found in FMHDRD, Items 10 and 11.

[##] FMCATF, FMCIDB, FMCTDB, FMRDRD, and FMRDRM.

C. Alt

Alt = $(R^2 + R_e^2 + 2RR_e \sin El)^{\frac{1}{2}} - R_e$, where R_e is earth radius.

D. R

1. Prior to 12 March 1971

$$R = [I_R(85) + I_R(86) + I_{11}(3)] - \Delta R$$

where

$I_R(85) + I_R(86)$ is the unambiguous range found in FMRDRM, Items 85 and 86

$I_{11}(3)$ is range bias found in Calibration Record FMRR11, Item 3

ΔR is tropospheric refraction correction

2. On or After 12 March 1971

$$R = [I_R(85) + I_R(86)] - \Delta R$$

E. $\dot{R}$[#]

$\dot{R}$ is found in FMRDRM, Item 84.

F. Az

1. Prior to 12 March 1971

$$Az = I_R(13) + I_C(3)$$

where

$I_R(13)$ is Az encoder angle found in FMRDRM, Item 13

$I_C(3)$ is Az bias found in Calibration Record FMAACC, Item 3

2. On or After 12 March 1971

$$Az = I_R(13)$$

[#] This $\dot{R}$ is computed by the ARTP and only approximates the true $\dot{R}$.

G. El

1. Prior to 12 March 1971

$$El = I_R(14) + I_C(5) - \Delta E$$

where

$I_R(14)$ is El encoder angle found in FMRDRM, Item 14

$I_C(5)$ is El bias found in FMAACC, Item 5

ΔE is tropospheric refraction correction

2. On or After 12 March 1971

$$El = I_R(14) - \Delta E$$

H. Az Offset

$$\text{Az offset (deg)} = \frac{\text{VHF Tr error (V)}}{\text{VHF LC sum (V)}} * \frac{1}{\Delta\text{Tr slope}} * \frac{1}{\cos El} * XK$$

where

VHF Tr error is found in FMRDRM, Item 19, if ALTAIR is in the point target tracking mode and in FMRDRM, Item 99, if ALTAIR is in the chaff tracking mode.#

VHF LC sum is found in FMRDRM, Item 23, if ALTAIR is in point target tracking mode and in FMRDRM, Item 104, if ALTAIR is in chaff tracking mode.

ΔTr slope is found in Calibration Record FMRR05, Item 2, if ALTAIR is in point target tracking mode and in Calibration Record FMR5CH, Items 2-6, if ALTAIR is in chaff tracking mode.

XK is a conversion factor from mrad to deg

#The mode is found in FMRDRM, Item 107.

J. El Offset

$$\text{El offset (deg)} = \frac{\text{VHF El error (V)}}{\text{VHF LC sum (V)}} * \frac{1}{\Delta\text{El slope}} * \text{XK}$$

where

VHF El error is found in FMRDRM, Item 20, if ALTAIR is in point target tracking mode and in FMRDRM, Item 100, if ALTAIR is in chaff tracking mode.

ΔEl slope is found in FMRR05, Item 4, if ALTAIR is in point target tracking mode and in FMR5CH, Items 12-16, if ALTAIR is in chaff tracking mode.

K. RCS#

$$\text{RCS} = 10 \log \left\{ \left[\frac{B_1 * R^2 * 2^{(\text{TGC+RGC})/6}}{B_5 * P^{\frac{1}{2}}} \right]^2 * I_R^2 \right\} + (10 * \text{KSENS})$$

where

B_1 is a calibration constant stored in the program. For VHF RCS, $B_1 = 0.460 * 10^{-13}$; for UHF RCS, $B_1 = 0.21 * 10^{-13}$

TGC is the TGC attenuation (db) found in FMRDRM, Item 26

RGC is the RGC attenuation (db) found in FMRDRM, Item 25

B_5 is a constant stored in the program as a function of waveform. It takes on the following values:

Waveform	Value
Long chirp	10
Short chirp	4.5
CW	1
CWL	10

P is peak power found in FMRDRM, Item 22 (VHF) and FMRDRM, Item 32 (UHF). It is used in counts.

#The RCS computed by ALTCEP should not be used for signature analysis since B_1 is a nominal calibration constant and is not measured each mission.

I_R is the frequency and polarization dependent target return found in FMRDRM, Item 23 (VHF LC), FMRDRM, Item 29 (VHF RC), FMRDRM, Item 33 (UHF LC), and FMRDRM, Item 39 (UHF RC). These items are used in counts.

KSENS is the receiver sensitivity bit found in FMRDRM, Item 51.

L. VHF Waveform[#]

VHF waveform is determined by combining FMRDRM, Items 3 and 28.

M. VHF Peak Power

VHF peak power (MW) is found in FMRDRM, Item 22.

N. VHF Attenuation[##]

VHF attenuation (db) = I_R (25) - (10 * KSENS)

where

I_R (25) is VHF RGC attenuation found in FMRDRM, Item 25

KSENS is the receiver sensitivity bit found in FMRDRM, Item 51

O. VHF PRF

VHF PRF = 10^6/CATF (5)

where

CATF (5) is the VHF PRI (μs) found in FMCATF, Item 5

P. UHF Waveform[†]

UHF waveform is found in FMRDRM, Item 38

Q. UHF Peak Power

UHF peak power (MW) is found in FMRDRM, Item 32.

[#] Called V M in listing.

[##] Called VHF GN in listing.

[†] Called U M in listing.

R. UHF Attenuation[#]

$$\text{UHF attenuation (db)} = I_R (35) - (10 * \text{KSENS})$$

where

I_R (35) is UHF RGC attenuation found in FMRDRM, Item 35

KSENS is the receiver sensitivity bit found in FMRDRM, Item 51

S. VHF Range Track Mode

VHF range track mode indication combines the range track mode found in FMRDRM, Item 62, and the RTC loss of signal indicator found in FMRDRM, Item 73. The loss of signal, shown as an L to the right of the track mode, indicates that the signal is either saturated or of insufficient amplitude.

T. Angle Track Mode

Angle track mode indication combines the angle track mode found in FMRDRM, Item 69, and the ATC loss of signal indicator found in FMRDRM, Item 74.

U. Chaff Tracking Gate Width

Chaff tracking gate width is found in FMRDRM, Item 107:

Code	Width (km)
0	0 (point target tracking mode)
1	2.5
2	5
3	10
4	20
5	40

V. Track Reference

The track reference, i.e., centroid (CN), leading edge (LE), or trailing edge (TE), is found in FMRDRM, Item 61.

[#] Called UHF GN in listing.

W. VHF-UHF Range Difference

VHF-UHF range difference is found in FMRDRM, Item 75.

X. UHF Range Track Mode

UHF range track mode combines the UHF lock-on indicator found in FMRDRM, Item 125, and the UHF loss of signal indicator found in FMRDRM, Item 126.

Y. Tag Time

Tag time (s) is found in FMCATF, Items 1 and 2.

Z. UHF PRF

$$\text{UHF PRF} = \frac{10^6}{\text{CATF }(6)}$$

where

CATF (6) is the UHF PRI (μs) found in FMCATF, Item 6.

AA. No. of ARS VHF Targets

No. of ARS VHF targets $= C_{21} - 1$

where

C_{21} is total no. of VHF targets being tracked, including VHF dummy target, found in FMCATF, Item 21.

BB. No. of ARS UHF Targets

No. of ARS UHF targets $= C_{11} - C_{21} - 1$

where

C_{11} is total no. of targets being tracked, including VHF and UHF dummy targets, found in FMCATF, Item 11.

CC. ARS Target No. of ALTAIR Tracked Target

The procedure for estimating which ARS target is the ALTAIR tracked target, for each record on the catalog tape, is:

1. Determine no. of FMRDRM data blocks (0, 2, 4, or 6) in the record from FMRDRD, Item 2.

2. Compute TAL and R of ALTAIR tracked target for each FMRDRM data block.

3. Determine no. of minor cycles (1 to 8) in the record from FMCATF, Item 10.

4. Compute TAL for each minor cycle.

5. Compare TAL's for all minor cycles with TAL's for all FMRDRM data blocks. Determine minor cycle and FMRDRM data block closest in time.

6. Obtain R for each ARS tracked target (identified in FMCIDB) from FMCTDB.

7. Select ARS target closest in R to ALTAIR tracked target.

DD. Priority

Priority is found in FMCIDB, Item 4.

III. OPERATION

A. Input

Start and stop times (GMT)

Skip time (s)

Scale parameters for plots

Lift-off time[#] (GMT)

A sample input is given in Appendix A.

CARD 1 (18A4)

(Col.)		
1-72	TITL	Title for listing and plots

CARD 2 [2 (2I3, F7.3), F7.3, 3I5, 2I3, F7.3]

1- 3	ISTART(1)	(h)	Start time (GMT)
4- 6	ISTART(2)	(min)	
7-13	TSTART	(s)	
14-16	ISTOP(1)	(h)	Stop time (GMT)
17-19	ISTOP(2)	(min)	
20-26	TSTOP	(s)	
27-33	DELTAT		Seconds between each line output in metric listing
34-38	IDELX[##]		Ordinate scale spacing (dbsm/in) (10)
39-43	MINXPL[##]		Minimum ordinate of plot (dbsm) (-50)
44-48	IDELT[##]		Abscissa scale spacing (s/in) (5) (If < 0, no plots are produced)
49-51	ILONCH(1)	(h)	Lift-off time
52-54	ILONCH(2)	(min)	
55-61	YLONCH	(s)	

[#] Normally left blank and lift-off time from Calibration Record FMGLOT is used.

[##] If left blank, program sets to indicated value.

B. Output

METRIC LISTING

GMT h, min, and s

TAL

Alt

R

$\dot{R}$ (m/s)

Az

El

Az offset

El offset

VHF LC RCS

UHF LC RCS

VHF waveform:[#] CW = CW; S = short chirp; L = long chirp; CL = CW long pulse

VHF power (dbw)

VHF attenuation[##] (db)

VHF PRF

UHF waveform[†]

UHF power (dbw)

UHF attenuation[††]

VHF range track mode: T = track; C = coast; L = lock-on; S = slaved

VHF range tracker loss of signal indication (L printed to right of track mode symbol)

[#] Called V M in listing.

[##] Called VHF GN in listing

[†] Called U M in listing.

[††] Called UHF GN in listing.

Angle track mode: T = track; C = coast; M = manual; S = slaved

Angle tracker loss of signal indication (L printed to right of angle mode symbol)

Chaff tracking gate width (km)

Track reference: TE = trailing edge; CN = centroid;
LE = leading edge

VHF - UHF range difference

UHF range track mode: T = track; blank = not in track

UHF range tracker loss of signal indication (L printed to right of track mode symbol)

ARS TARGET LISTING

GMT h, min, and s

TAL

Tag time (s)

VHF PRF

UHF PRF

No. of VHF targets

No. of UHF targets

ARS target no., frequency, and priority for target closest in range to the ALTAIR tracked target

Absolute range difference between selected ARS target and the ALTAIR tracked target

ARS target test data appear only when one of the following items change: VHF PRF, UHF PRF, no. of VHF targets, no. of UHF targets, ARS target no., frequency, and priority.

PLOTS

RCS plots vs TAL of ALTAIR tracked target
(VHF and UHF LC and RC)

Sample outputs are presented in Appendix B.

IV. PROGRAM LIMITATIONS

Start time	Must be before end of tape
Stop time	Must be before start of tape
DELTAT	Must be a multiple of 0.025 s

V. PROGRAMMING

A. ALTCEP (see Appendices C and D.)

ALTCEP is the control section of ALTCEP. ALTCEP reads the input cards, calls the subroutines, and prints the data.

B. CHEAD (see Appendices E and F.)

CHEAD is used to process calibration and format records which are recorded before the data records on the ALTAIR transcription tape. CHEAD lists format and calibration records named in the common statement. These are stored and unpacked for later use by the main processing program. A sample CHEAD output is given in Appendix G.

The minimum size needed for the item array may be calculated by the following equation:

Item size = 6 * (Total number of items stored) + (6 * 130)

Calling Sequence: Call CHEAD (*)

* = A return point specified by a statement number in the calling program. Used for aborting job by main program if wrong tape is mounted.

CHEAD calls the following subroutines:

BREADS (entries BREADS and BREAD); HDRR (entries HDRR and NAMET); and FORM.[2] CHEAD also calls WHICHV, a 360-system subroutine indicating whether a job is being run under the time-sharing (CMS) or Batch (OS) systems.

C. GET and IGET

GET and IGET are entries to subroutine GETS.[2]

These routines will locate any data item, unpack it, and interpret it according to the information in the format table. They will return the item as a binary integer (in the case of IGET) or as a floating point number (in the case of GET).

GET (or IGET) requires three arguments:

GET (Format, Base, Item)

Format	Relevant format table address
Base	Base address of data block desired
Item	Specific item number

D. LTIME (see Appendix H.)

Subroutine LTIMF unpacks the lift-off time. The call statement is LTIME (LOT, IH, IM, IS, IT).

INPUT

LOT	Unpacked lift-off time

OUTPUT

IH	h
IM	min
IS	s
IT	ms

E. GMTTAL, CATIME, and RADART (see Appendix J.)

GMTTAL, CATIME, and RADART are entries to Subroutine GMTTAL.

1. GMTTAL

Entry GMTTAL converts start time and stop time on input data cards to total GMT s and computes TAL. The call statement is GMTTAL (IGMTH, IGMTM, GMTS, TAL).

INPUT

IGMTH	Hours
IGMTM	Minutes
GMTS	Seconds and decimal fractions of seconds

OUTPUT

TAL	Time (s) after launch

STORED IN COMMON

TLONCH	Lift-off time (total GMT s)

2. CATIME

Entry CATIME obtains GMT from FMCATF, unpacks and converts it to total GMT s, and computes TAL. The call statement is CATIME (FMCATF, IADD, IGMTH, IGMTM, GMTS, TAL).

INPUT

FMCATF	FMCATF format table address
IADD	Base address of FMCATF data block

OUTPUT

IGMTH	Hours
IGMTM	Minutes
GMTS	Seconds and decimal fractions of seconds
TAL	Time (s) after launch

STORED IN COMMON

TLONCH	Lift-off time (total GMT s)

3. RADART

Entry RADART obtains GMT from FMRDRM, unpacks and converts it to total GMT s, and computes TAL. The call statement is RADART (FMRDRM, IRDRM, IGMTH, IGMTM, GMTS, TAL).

INPUT

FMRDRM	FMRDRM format table address
IRDRM	Base address of FMRDRM data block

OUTPUT

IGMTH	Hours
IGMTM	Minutes
GMTS	Seconds and decimal fractions of seconds
TAL	Time (s) after launcn

STORED IN COMMON

TLONCH	Lift-off time (total GMT s)

F. REFC (see Appendix K.)

The tropospheric refraction correction subroutine, REFC, is based on tropospheric refraction tables in PPP-36.[3] A modified version of this subroutine is now in use.

The call statement is REFC (E, R, DEE, DRR)

E	Uncorrected El (must be between 0° and 90°)
R	Uncorrected R (ft)
DEE	El tropospheric correction
DRR	R tropospheric correction (ft)

The corrected values to be computed after exiting from REFC are:

$$\mathrm{El} = \mathrm{E} - \mathrm{DEE}$$
$$\mathrm{R}\ (\mathrm{ft}) = \mathrm{R} - \mathrm{DRR}$$

G. BZERO (see Appendix L.)

Subroutine BZERO is necessary if a floating point item is scaled B0 in a format or calibration table description. BZERO is called after the item has been extracted by subroutine GETS.[2] BZERO normalizes the data item and puts the decimal point in its proper position. The call statement is BZERO (yy).

INPUT AND OUTPUT

yy	Item to be processed

H. IBIT (see Appendix M.)

Function IBIT identifies which bit is set from a group of bits, assuming one and only one bit is set. For example, track reference is determined from FMRDRM, Item 61. There are three bits of binary data, numbered from right to left, in the the format ABC where

A indicates leading edge

B indicates centroid

C indicates trailing edge

The call statement is IBIT (ITEM, NBIT).

INPUT

ITEM	Designation of group of bits to be searched
NBIT	No. of bits in group

OUTPUT

No. of the bit that is set

J. METAZ

METAZ is the plotting routine.

K. Plotting System Subroutines

They are PLTND, STOIDV, and REREAD.

REFERENCES

1. "ALTAIR Data User's Manual", LM-97, Lincoln Laboratory, M.I.T. (to be published), UNCLASSIFIED.

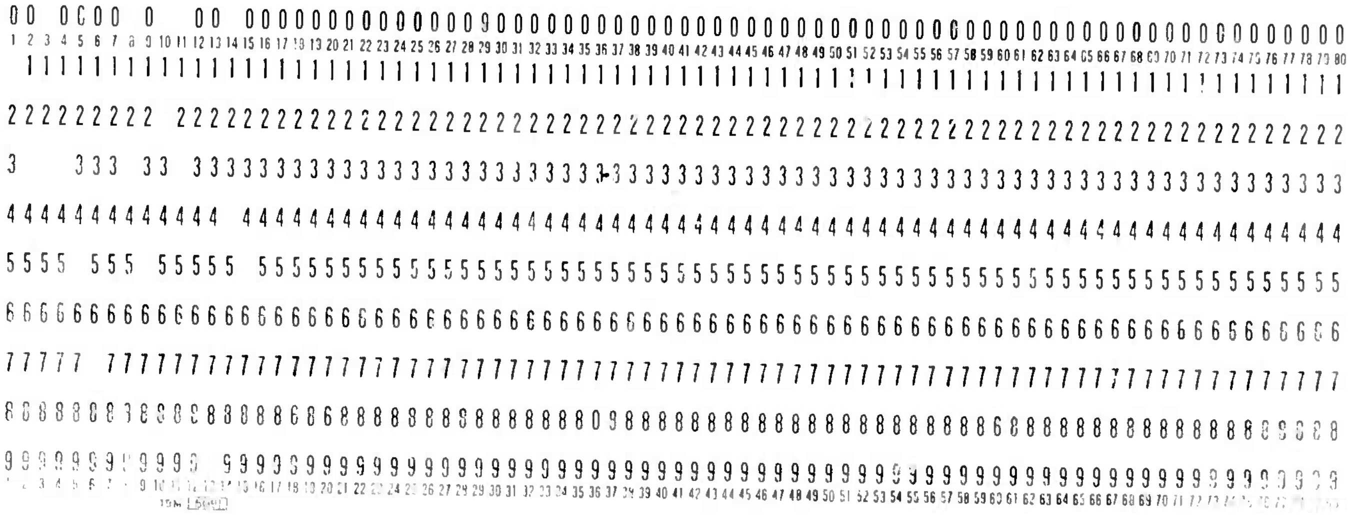
2. "Data Reduction Program Documentation, ALTAIR Tape Read Package, (Effective: April 1970)", PA-229-1, Lincoln Laboratory, M.I.T. (17 March 1971), UNCLASSIFIED.

3. J. P. Penhune, "Refraction Corrections for the TRADEX Radar", PPP-36, Lincoln Laboratory, M.I.T. (21 April 1965), UNCLASSIFIED.

APPENDIX A
ALTCEP INPUT

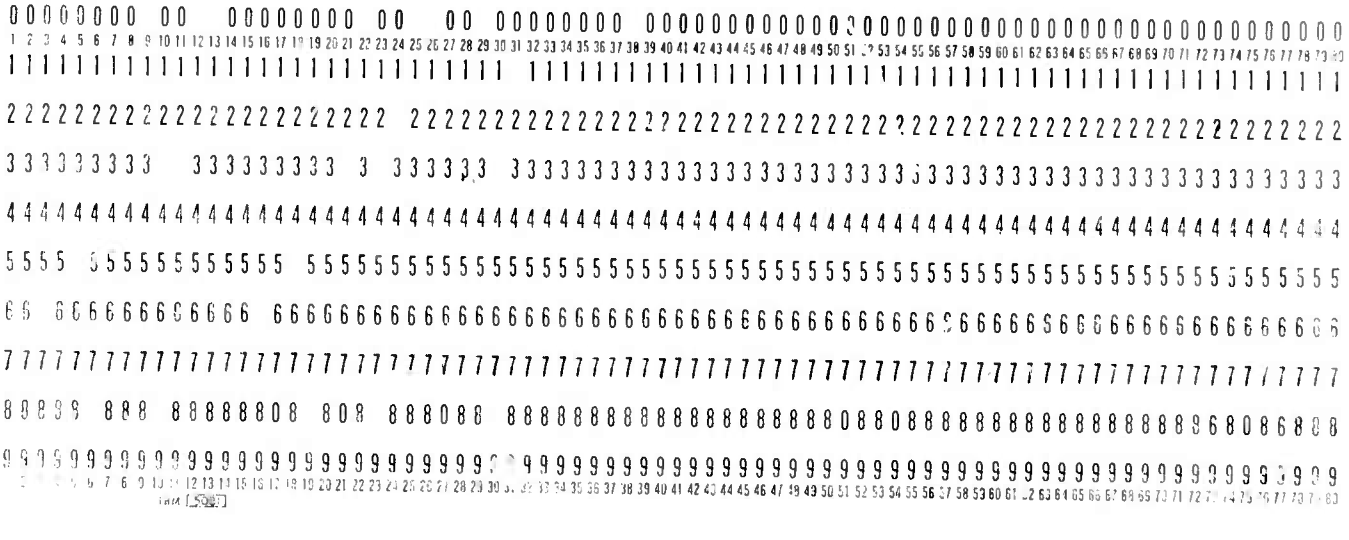
ALTCEP TEST RUN

CARD 1



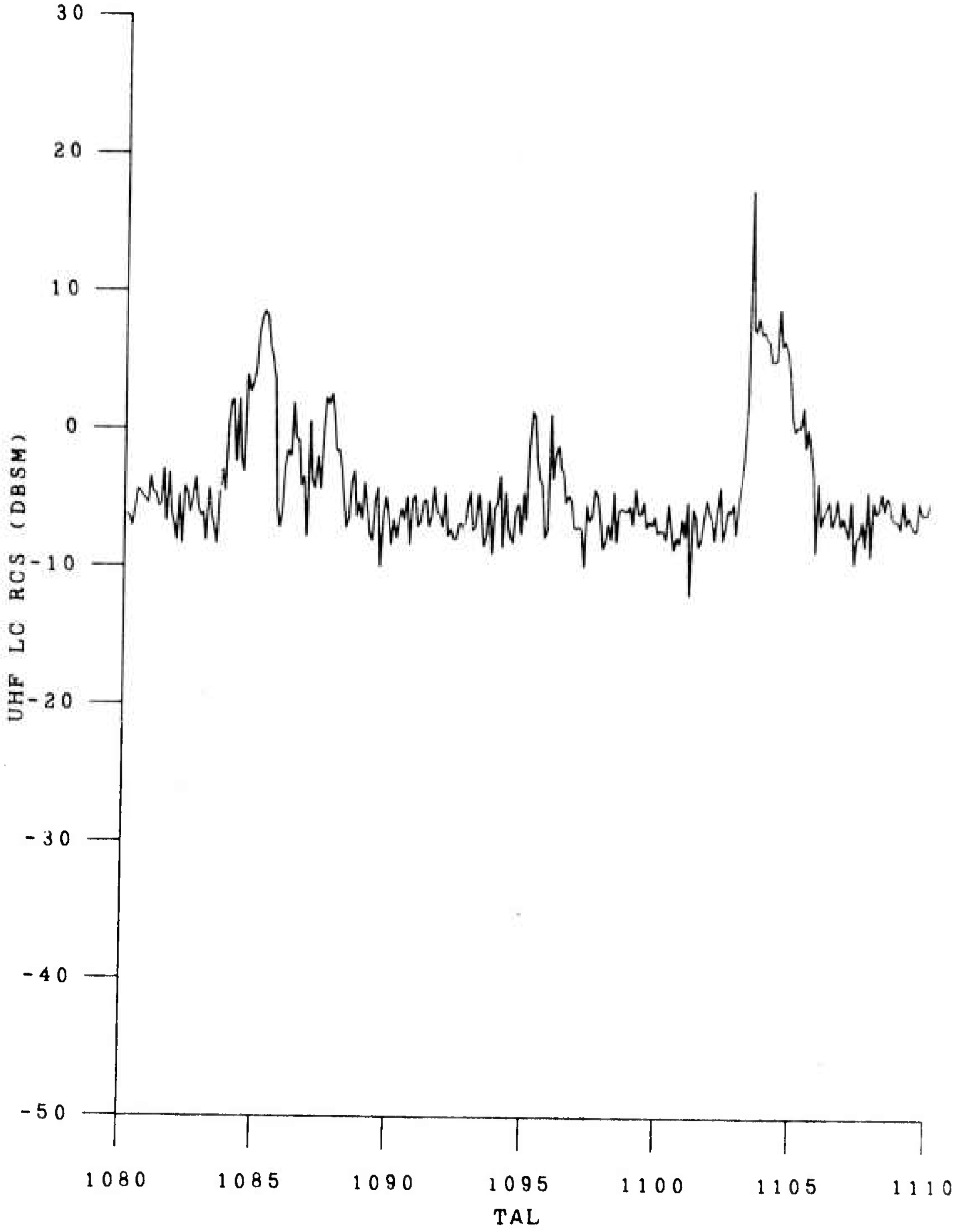
6 58 0.300 6 58 30.200 0.1 0

CARD 2



APPENDIX B
ALTCEP OUTPUTS

```
24000.203     6:40: 0.207 LAUNCH

1PROC= 0

ARTP VERSION DATE = 12MAR71
RTS VERSION DATE = 11MAR71

START  6 58  0.300 STOP  6 58 30.200 DELTA  0.100 MIN    0 DELTA X  -10 DELTA T    0 LAUNCH  6 40  0.207

ZEMKE ALTCEP NEW                                                                                          PAGE    1
    GMT          TAL      ALT   RANGE    RANGE     AZ    EL    AZ    EL     VHF    UHF  V  VHF VHF VHF U  UHF UHF  MODE  CHF TRK VR     UHF
                                         RATE                  OFF   OFF    LC     LC   M  PWR  GN PRF M  PWR  GN         GTE REF  -UR   RTK
                (SEC)    (KM)   (KM)     (M/S)     (D)   (D)   (D)   (D)  (DBSM) (DBSM)   (DBW) (DB)     (DBW) (DB) R   A (KM)      (KM)

6 58  0.300 1080.097 1173.0 3454.9 -4854.4   58.6  5.7 -0.53 -0.46   0.4   -6.2 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.40  T
6 58  0.400 1080.197 1172.9 3454.4 -4854.6   58.6  5.7 -0.12  0.08   0.3   -6.4 CL 59.1   0  186 L 70.6   0 CL CL 10.0 CN -1.40  T
6 58  0.500 1080.297 1173.0 3453.9 -4854.9   58.6  5.7 -0.02 -0.14  -3.1   -7.0 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.41  T
6 58  0.600 1080.397 1172.8 3453.4 -4855.2   58.6  5.7 -0.24 -0.30   5.9   -5.8 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.41  T
6 58  0.700 1080.497 1172.6 3453.0 -4855.4   58.6  5.7 -0.31 -0.14  -1.2   -4.4 CL 69.1   0  186 L 70.6   0 CL CL 10.0 CN -1.42  T
6 58  0.800 1080.597 1172.8 3452.5 -4855.7   58.5  5.8 -0.24  0.06   1.5   -4.6 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.42  T
6 58  0.900 1080.697 1172.5 3452.0 -4855.9   58.6  5.8 -0.41 -0.18   2.2   -4.9 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.43  T
6 58  1.000 1080.797 1172.6 3451.5 -4856.2   58.6  5.8 -0.21 -0.43   3.9   -5.1 CL 69.1   0  186 L 70.6   0 CL CL 10.0 CN -1.43  T
6 58  1.100 1080.897 1172.6 3451.0 -4856.4   58.6  5.8 -0.43 -0.08  -1.5   -5.4 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.44  T
6 58  1.200 1080.997 1172.5 3450.5 -4856.7   58.6  5.8  0.12 -0.20   3.1   -3.5 CL 69.1   0  186 L 70.6   0 CL CL 10.0 CN -1.44  T
6 58  1.300 1081.097 1172.3 3450.0 -4856.9   58.6  5.8 -0.16 -0.04   5.3   -4.6 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.45  T
6 58  1.400 1081.197 1172.1 3449.6 -4857.2   58.6  5.8  0.0   0.53   3.2   -4.7 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.45  T
6 58  1.500 1081.297 1172.2 3449.1 -4857.4   58.6  5.8 -0.21 -0.02   2.0   -5.6 CL 69.1   0  186 L 70.6   0 CL CL 10.0 CN -1.46  T
6 58  1.600 1081.397 1172.1 3448.6 -4857.7   58.5  5.8 -0.36 -0.41  -2.8   -5.5 CL 69.2   0  186 L 70.6   0 CL CL 10.0 CN -1.46  T


      ZEMKE ALTCEP NEW
          GMT           TAL      TAG TIME     V PRF     U PRF    NO. TARG.        POSSIBLE  SYLVANIA  TARGET
                       (SEC)      (SEC)                          VHF   UHF    ARS TARG.   PRIOR.   RANGE DIFF. (KM)

      6 58  0.357    1080.154    561.482      186       186      0     1
      6 58 28.261    1108.059    589.385      359       359      0     1        2  U        4          4.30
```


30
20
10
0
-10
-20
-30
-40
-50
UHF LC RCS (DBSM)
1080
1085
1090
1095
1100
1105
1110
TAL

APPENDIX C

ALTCEP PROGRAM LISTING

```
C
      COMMON/8EAD/LN,IFLG,IADD,FMRDID,FMCATF,FMCSAD,FMCMD8,FMCTI8,FMCIDB
     1,FMCTDB,FMRDRD,FMRDRM,FMRDRT,FMGLOT,FMRR05,FMAXSP,FMBIAS,FMR5CH
     2,FMRCHF,FMAACC,FMRR11,NAME(19),NI(18),IX(18),ITEM(8000)
      COMMON/TITLE/IDARR(10)
      COMMON /LAUNCH/ TLONCH
C
      INTEGER * 2 ITEM,TKREF1,TKREF,L8IT,ITLBIT,IRTBIT
      INTEGER * 2 I25MIL, IAM8, I8UFF, I8UFO, IFLAG, IGMTH, IGMTH2,
     1 IGMTM, IGMTM2, ILONCH, IPAGE, IPRI, IPRIO2, ISTART, ISTOP, ITAR2,
     2 ITARG, ITKMOD, ITYPE2, IUPRF2, IVPRF2, JTYPE, KFLAG, KCN10, LCS,
     2 LOSANG, LOSRAN, MAX8UF,IGATE,
     3MAXL12, MAXLIN, MFLAG, MINOR, MODE, MODTK1, MODTK2, N25MIL,
     4NCHECK, NFLAG, NRDRM, NTGRN, NTGRO, NTCT, NTOTE, NTYPE, NUHF2,
     5NUHFO, NUMBUF, NUMLIN, NUMREC, NUMUHF, NUMVHF, NVHF2, NVHFC,
     6UGAIN, UMODE, VGAIN, VMODE
      INTEGER * 2 NYR(2),MTH1(2),MTH2(2),MTH3(2),MTH4(2),NDAY1(2)
     1,MTH(2),IYR(2),MTHA(2),NDAY(2),LA8YAX(9,4)
C
      REAL * 8 GMTS, GMTS2, TAGTI2, TAL, TAL2, TIME, TMINOR, TRTAL2,
     1TSTART, TSTOP
      REAL * 8 TLONCH
C
      DIMENSION CROSFC(1210,4), DRANG2(570), GATSEL(6), GMTS2(570),
     1 IAMB(20),LBIT(3),
     1IAX(2), I8UF1(2048), IBUF2(2048), I8UFF(570,7), I8UFO(7),
     2IGMTH2(570), IGMTM2(570), ILONCH(2), INSTR(16), IPRI(20),
     3 IPRIO2(570), ISTART(2), ISTOP(2), ITAR2(570),
     3ITARG(20),IXTICK(9),KXTICK(7),JXTICK(9),ITICK(9),ILCT(3),
     4ITIME(1210), ITKMOD(2), ITYPE2(570), IUPRF2(570), IVPRF2(570),
     5 IXPOS(2), IYPOS(2), JAX(2), JCROSS(1210,4), JTICK(9), JTYPE(3),
     6KAX(2),KTICK(9),LA8EL(16),LAX(2),LOCX(16),
     7 LOCY(16), LOS(2), MESAGE(18), MODE(5), MODTK1(4), MCDTK2(4),
     7 NTYPE(256),
     8NUHF2(570), NVHF2(570), PLOTIM(1210), RANGE2(10), TAGTI2(570),
     9 TAL2(570), TRTAL2(10), XLEFT(1210), XRITE(1210), XXMES(12),
     A YLEFT(1210), YRITE(1210)
       DIMENSICN MCATF(8), TKREF1(3)
C
      EQUIVALENCE (IBUFF(1,1), IVPRF2(1)), (I8UFF(1,2), IUPRF2(1)),
     1(I8UFF(1,3), NVHF2(1)), (IBUFF(1,4), NUHF2(1)), (IBUFF(1,5),
     2ITAR2(1)), (I8UFF(1,6), ITYPE2(1)), (IBUFF(1,7), IPRIO2(1))
      EQUIVALENCE (MTNXPL, LABEL(8)), (PLOTIM(1), ITIME(1)),
     1 (CROSEC(1,1), JCROSS(1,1), XLEFT(1)), (CROSEC(1,2), XRITE(1)),
     2 (CROSEC(1,3), YLEFT(1)), (CROSEC(1,4), YRITE(1))
      EQUIVALENCE (FMCATF, MCATF(1))
      EQUIVALENCE (MYR,NYR(1)),(NTH,MTH(1)),(NTHA,MTHA(1)),
     1(MDAY,NDAY(1))
C
      DATA IYR/'  71'/,MTH1/'  JA'/,MTH2/'  FE'/,
     1M/Z5FFFFFFF/,MTH3/'  MA'/,MTH4/'  OR'/,KK/16777216/
     2,L/Z7FFFFFFF/,NDAY1/'  12'/,MM/Z0000F080/,NN/Z00008000/
      DATA CDN10 /10./, CON30 /30./, CONALT /.30480061/, CCNDEG /360./,
     1CONOFP /.078125/, CONPDW /1.E-10/, CONRAN /.00091440183/,
     2CCNRAT /.0027905329/, CONRCS /.5511811C24/, CONUP /2.66666667/,
     3 CCNVP /1.333333333/, GATSEL /0., 2.5, 5., 10., 20., 40./,
     3 I2T22 /4194304/,IAX/1020,1020/,IFL/0/,
     5ITICK / 9* 400/, IXPOS/ 1911,770 /,IYPCS/ 175,1309/,
     6JAX/400,400/,JTICK/9*325/,JTYPE/'  ',' V',' U'/,
     7KAX /2880,1020/, KON10 /10/, KONTAG /1C00000/, KTICK /400, 710,
     81020, 1330, 1640, 1950, 2260, 2570, 2880/, LABXAX /'TAL '/
      DATA LABYAX/'VH','F ','LC',' R','CS',' (','DB','SM',') ','VH',
     1'F ','RC',' R','CS',' (','DB','SM',') ','UH','F ','LC',' R','CS',
     2' (','DB','SM',') ','UH','F ','RC',' R','CS',' (','DB','SM',') '/
      DATA LAX/400,2880/,LOCX/963,1273,1583,1893,2203,2513,2823,
     19 * 775/, LOCY /7 * 250, 400, 710, 1020, 1330, 1640, 1950, 2260,
     2 2570, 2880/, LOS /'  ', 'L '/, MAX /8192/, MAXBUF /570/,
     3 MAXLIN /55/, MFLAG /1/, MODE /'CW', ' S', ' L', '  ', 'CL'/,
     4 MODTK1 /'T ', 'C ', 'L ', 'S '/, MODTK2 /'T ', 'C ', 'M ', 'S '/,
     4INSTR /' OPERATOR - USE 11 X 14 HARD CCPY LENS AND CUT. THANKS'/,
     5 NFLAG /1/, NTGRO /0/, NTYPE /256 * 0/, NUHFO /0/, NUMBUF /1/,
     6 NVHFO /0/, TKREF1 /'TE','CN','LE'/,
     7KXTICK/1020,1330,1640,1950,2260,2570,2880/,IXTICK/9*1020/,
     8JXTICK/9*945/
      DATA ICMS /'CMS '/,LBIT/'  ','T ','L '/
C
 6000 FORMAT (2(2I3, F7.3), F7.3, 3I5,2I3,F7.3)
 6001 FORMAT(2(2I3,F7.3),F7.3,2I3,F7.3)
6010  FORMAT(6X,'GMT',8X,'TAL',5X,'ALT',2X,'RANGE',3X,'RANGE',4X,'AZ',
     13X,'EL',3X,'AZ',4X,'EL',4X,'VHF',4X,'UHF',2X'V',2X,'VHF',1X,
     2'VHF',1X,'VHF',2X,'U',2X'UHF',1X,'UHF',2X,'MODE',2X,'CHF',
     31X,'TRK',1X,'VR',3X,'UHF'/38X,'RATE',15X,'OFF',3X,'CFF',3X,
     4'LC',5X,'LC',3X,'M',2X,'PWR',2X,'GN',1X,'PRF',2X,'M',2X,'PWR',
     52X,'GN',8X,'GTE',1X,'REF',2X,'-UR',1X,'RTK'/16X,'(SEC)',3X,
     6'(KM)',2X,'(KM)',3X,'(M/S)',4X,'(D)',2X,'(D)',2X,'(C)',
     73X,'(D)',2X,'(DBSM)',1X,'(DBSM)',2X,'(DBW)',1X,'(DB)',6X,
     8'(DBW)',1X,'(DB)',1X,'R',2X,'A',1X,'(KM)',5X,'(KM)'/)
6020  FORMAT(2I3,F7.3,F9.3,2F7.1,F8.1,F6.1,F5.1,2F6.2,2F6.1,1X,A2,F5.1,
     1I4,I5,A2,F5.1,I4,2(1X,2A1).F5.1,1X,A2,1X,F5.2,2X,2A1)
 6030 FORMAT (     6X, 'GMT', 10X, 'TAL', 5X, 'TAG TIME', 5X, 'V PRF',
     14X, 'U PRF', 4X, 'NO. TARG.', 9X, 'POSSIBLE  SYLVANIA  TARGET' /
     218X, '(SEC)', 6X, '(SEC)', 24X, 'VHF', 3X, 'UHF', 4X, 'ARS TARG.',
     33X, 'PRIOR.', 3X, 'RANGE DIFF. (KM)' /)
 6040 FORMAT (2I3, F7.3, 2F11.3, 2I9, I7, I6, I9, 1X, A2, I8, F14.2)
 6050 FORMAT (//'   UP TO', I5, ' RECOROS SKIPPED'//)
 6060 FORMAT (1H1)
 6070 FORMAT (' START',2I3,F7.3, ' STOP',2I3,F7.3,' DELTA',F7.3,
     1' MIN',I5,' DELTA X',I5,' OELTA T',I5,' LAUNCH',2I3,F7.3)
 6080 FORMAT (16A4)
 6090 FORMAT (16I4)
 6100 FORMAT (18A4)
 6110 FORMAT (1H1, 18A4, 48X, 'PAGE', I5)
C
      IGLOT(K) = IGET(FMGLOT,IADD,K)
      RR11(K) = GET(FMRR11,IADD,K)
      AACC(K) = GET(FMAACC,IADD,K)
      CATDAT(K) = GET(FMCATF,IADD,K)
      ICADAT(K) = IGCT(FMCATF,IADD,K)
      IFORD(K) = IGET(FMRDRD,IRDRD,K)
      IRCDAT(K) = IGFT(FMRDRM,IRDRM,K)
      RDDAT(K) = GET(FMRDRM,IRDRM,K)
      IDBCAT(K) = IGFT(FMCIDB, IDBASE, K)
      C2(K) = GET(FMR5CH,IADD,K)
C
```

```
      CALL WHICHV(ID)
      WRITE(6,1) ID
  1   FORMAT(' ID=' A4)
      IOUT=6
      IF(ID.EQ.ICMS)IOUT=8
C
      XK=.05729578
      B1UHF = 0.2099581E-13
      B1VHF = 0.4604345E-13
      KFLAG = 1
      IPAGE = 0
      CALL CHEAD (&147)
C
      READ (5, 6100) MESAGE
      READ(5,6000) ISTART(1),ISTART(2),TSTART,ISTCP(1),ISTOP(2),TSTOP,
     1DELTAT,IDELX,MINXPL,IDELT,ILONCH(1),ILCNCH(2),YLONCH
      IF((ILCNCH(1).NE.0).OR.(ILONCH(2).NE.0).OR.(YLONCH.NE.0.))
     1 GO TO 21
      DO 11 I=1,3
      ILOT(I) = IGLOT(I)
 11   CCNTINUE
      TLCNCH = 0.0
      IF(ILOT(1) .NE. 2) GO TO 22
      CALL LTIME (ILOT, NH, NM, NS, NT)
      ILONCH(1) = NH
      ILCNCH(2) = NM
      YLCNCH = NS + FLOAT(NT) * .001
      TLCNCH = ILONCH(1) * 3600 + ILCNCH(2) * 60 + YLONCH
      PRINT 23,TLONCH,ILONCH(1),ILONCH(2),NS,NT
23    FORMAT('1'F9.3,3X,2(I2,'0'),I2,'.'I3,' LAUNCH')
      GO TO 22
   21 TLCNCH=ILONCH(1)*3600+ILONCH(2)*60+YLONCH
22    CCNTINUE
C
      N = IDARR(10)
      J = IDARR(9)
      J1 = MASK(J,L)
      N1 = MASK(N,M)
      J3 = ICARR(7)
      N3 = IDARR(8)
      IPROC = 0
      MYR = MOD(N1/256,65536)
      NTH = MOD(J1,65536)
      N2 = MCD(N1/KK,256)
      NTHA = MAKE1(N2,MM)
      MDAY = MCD(J1/65536,65536)
      MDAY = MAKE1(MDAY,NN)
      IF(NYR(2).LT.IYR(2))GO TO 12
      IF(NYR(2).GT.IYR(2)) GO TO 13
      IF(MTH(2).EQ.MTH1(2).OR.MTH(2).EQ.MTH2(2))GC TO 12
      IF(MTH(2).NE.MTH3(2))GO TO 13
      IF(MTHA(2).NE.MTH4(2))GO TO 13
      IF(NDAY(2).LT.NDAY1(2)) IPROC = 1
      GO TO 13
12    IPROC = 1
13    CCNTINUE
      PRINT 991,IPROC,J,N,J3,N3
```

```
991   FORMAT(//' IPROC=',I2//' ARTP VERSION DATE = ',2A4/,
     1' RTS VERSION DATE = ',2A4//)
C
C INITIALIZE PLOTTING
C
      NPLOT = 0
      PLMINX = -50.
      IF (IDELX) 6, 3, 2
    2 PLMINX = MINXPL
      GO TO 6
    3 MINXPL = -50
      IDELX = 10
      IDELT = 5
6      WRITE (6,6070) ISTART(1),
     1ISTART(2), TSTART, ISTOP(1), ISTOP(2), TSTOP, DELTAT, MINXPL,
     2IDELX, IDELT,ILONCH(1),ILONCH(2),YLONCH
C
      IF (IDELX .LT. 0) GO TO 8
      DELSIX = 6. * IDELT
      CALL STOIDV(MESAGE,71,0)
      CALL REREAD (99)
      CALL METAZ (20, 0, 0, 0, 0, 0)
      CALL METAZ (50,24,2,0,0,0)
      CALL METAZ (51,35,0,0,0,0)
      CALL METAZ (51, 36, 0, 0, 0, 0)
      CALL METAZ (51, 37, 4095, 0, 0, 0)
      CALL METAZ (51, 38, 3071, 0, 0, 0)
      CALL METAZ (51, 31, 0, 0, 0, 0)
      CALL METAZ (51, 32, 512, 0, 0, 0)
      CALL METAZ (51, 33, 4095, 0, 0, 0)
      CALL METAZ(51, 34, 3583, 0, 0, 0)
      CALL METAZ (50, 5, 2, 0, 0, 0)
      CALL METAZ (51, 27, 39, 0, 0, 0)
      CALL METAZ (50, 3, 1, 0, 0, 0)
      CALL METAZ (5, 0, 720, 1600, 72, MESAGE(1))
      WRITE (99,6001) ISTART(1),ISTART(2),TSTART,ISTOP(1),ISTOP(2),
     1TSTOP,DELTAT,ILONCH(1),ILONCH(2),YLONCH
      READ (99, 6080) XXMES
      CALL METAZ (5, 0, 720, 1400, 48, XXMES(1))
      CALL METAZ (13, 1, 0, 0, 0, 0)
    8 CONTINUE
C
      IF((ISTART(1).EQ.0).AND.(ISTART(2).EQ.0).AND.(TSTART
     1.EQ. 0.) ) GO TO 10
      CALL GMTTAL (ISTART(1), ISTART(2), TSTART, TSTART)
C
   10 IF ( (ISTOP(1) .NE. 0) .OR. (ISTOP(2) .NE. 0) .OR. (TSTOP .NE. 0)
     1) GO TO 20
      TSTOP = 1000000.
      GO TO 25
   20 CALL GMTTAL (ISTOP(1), ISTOP(2), TSTOP, TSTOP)
C
   25 TIME = TSTART
      LN = 1
      NUMLIN = MAXLIN
      MAXLI2 = MAXBUF / 10
C
```

```
C DETERMINE TICK LABELS
C
      IF (IDELX .LT. 0) GO TO 32
      J = TSTART / IDELT
      LABEL(1) = J * IDELT
      DO 28 I = 2, 7
      LABEL(I) = LABEL(I-1) + IDELT
   28 CONTINUE
      TIMMIN = LABEL(1)
      TIMMAX = LABEL(7)
      DO 30 I = 9, 16
      LABEL(I) = LABEL(I-1) + IDELX
   30 CONTINUE
      PLMAXX = LABEL(16)
      RCSDIF = PLMAXX - PLMINX
      WRITE (99, 6090) LABEL
      READ (99, 6080) LABEL
   32 CONTINUE
      IF (MCATF(1) .EQ. 0) CALL EXIT
      DO 33 I = 5,8
      IF (MCATF(I) .EQ. 0) CALL EXIT
  33  CONTINUE
C
   35 CALL BREADS (LN, IBUF1, IBUF2, MAX, IFL, INDX, LEN, IFLG, IADD)
      NUMREC = 0
   40 CALL BREAD (LN)
      GO TO (60, 45, 50, 50), IFLG
   45 NUMREC = NUMREC + 1
      GO TO 40
   50 NFLAG = 2
      MFLAG = 2
      NUMBER = NPLOT
      IF (NUMREC .EQ. 0) GO TO 142
      IF(NUMLIN.GT.50) WRITE(IOUT,6060)
      WRITE(IOUT,6050) NUMREC
      GO TO 142
C
   60 IRORD = ICADAT(19) * 3 + IADD
      IRORD2 = IRORD
      IREL = (IFORD(1) - 1) * 3
      N25MIL = IFORD(2)
      I25MIL = IFORD(3) * 3
      IREL = 0
      N25MIL = 2
      I25MIL = 120
C
      CALL CATIME (MCATF, IADD, IGMTH2(NUMBUF), IGMTM2(NUMBUF),
     IGMTS2(NUMBUF), TAL2(NUMBUF))
      IVPRF2(NUMBUF) = KONTAG / ICADAT(5)
      IUPRF2(NUMBUF) = KONTAG / ICADAT(6)
C
      GO TO (70,130), NFLAG
   70 IRORM = IRORD + IREL
      IF (IRORM .EQ. IREL) GO TO 130
      I50MIL = ICADAT(13) * 3
      NRORM = ICADAT(12) * N25MIL
      K25MIL = 0
```

```
C
      DO 120 I = 1, NRDRM
      CALL RADART(FMRDRM,IRDRM,IGMTH,IGMTM,GMTS,TAL)
      IF(TIME.EQ.0.0)TIME=TAL
      IF (DABS(TAL2(NUMBUF) - TAL) .LE. 1.) GO TO 72
      NUMREC = NUMREC + 1
      GO TO 40
   72 IF(((TIME-.0199) .LT.TAL).AND.((TIME+.01).GE.TAL)) GC TC 73
      IF((TIME - .01).GT.TAL) GO TO 110
      TIME = TIME + DELTAT
      GO TO 72
   73 CCNTINUE
      IF (TAL .LE. TSTOP) GO TC 75
      NFLAG = 2
      NUMBER = NPLOT
      IF (NUMREC .EQ. 0) GO TO 130
      IF(NUMLIN.GT.50) WRITE(IOUT,6060)
      WRITE(IOUT,6050) NUMREC
      GO TO 130
   75 DELTRA = IRDDAT(75) * 0.91440183/1000.
      IRTBIT = IRDDAT(125)+ 1
      IRTBIT = LBIT(IRTBIT)
      JTLBIT = IRCDAT(126)
      ITLBIT = LBIT(JTLBIT + 2)
      IF(IRDDAT(126).EQ.0)ITLBIT = LBIT(JTLBIT + 1)
      IF(IPROC.EQ.1) GC TO 751
       RANGE = (RDDAT(85) + RDDAT(86)) * 3.0
      RANRAT = RDDAT(84) * CONRAT
      EL = RDDAT(14) * CONDEG
      AZ = RDDAT(13) * CONDEG
      GO TO 752
  751 RANGE = (RDDAT(85)+RDDAT(86))*3.0+RR11(3)*1.8737*3.2808
      RANRAT = RDDAT(84) * CONRAT
      AZ = RDDAT(13)*CCNDEG+AACC(3)*57.295
      EL = RDDAT(14)*CONDEG+AACC(5)*57.295
  752 CALL REFC (EL, RANGE, DELEL, DELRNG)
      RANGE = (RANGE - DELRNG) * .0003048
      EL = EL - DELEL
      EL = EL*.0174533
      ERAD = 6378.145
      ALT = SQRT(RANGE**2+ERAD**2+2*RANGE*ERAD*SIN(EL))
      ALT = ALT-ERAD
      EL = EL/.0174533
       IGATE=IRDDAT(107)
      GATE = GATSEL(IGATE + 1)
      B2=IRDDAT(104)
      XB2=IRDDAT(23)
      IF(XB2 .NE. 0) GC TO 77
      AZOFF = 10.
      ELOFF = 10.
      GO TO 78
   77 IF(IGATE .NE. 0) GO 10 771
      X4 = GET(FMRRO5,IADD,4)
      X2 = GET(FMRRO5,IADD,2)
      CALL BZERO(X4)
      CALL BZERO(X2)
      AZOFF=(IRDDAT(19)/(XB2*COS(EL*.0174533)))*(XK/X2)
      ELOFF=(IRDOAT(20)/XB2)*(XK/X4)
      GO TO 772
771   IF (B2 .NE. 0) GO TO 773
      AZOFF = 10.
      ELOFF = 10.
      GO TO 78
773   XC2=C2(IGATE+1)
      CALL BZERO(XC2)
      AZOFF=(IROOAT(99)/(B2*COS(EL*.0174533)))*(XK/XC2)
       XC4=C2(IGATE+11)
      CALL BZERO(XC4)
      ELOFF=(IRDDAT(100)/B2)*(XK/XC4)
  772 CCNTINUE
78     IF (IRODAT(3) .EQ. 0) GO TO 781
      VMOOE - MODE(5)
      GO TO 782
  781 VMOOE = IRDDAT(28)
      VMOOE = MOOE(IBIT(VMOOE,4))
 782  VPOWER = RDOAT(22) * CONVP * CONOFP
      IF (VPOWER .LE. C.) VPOWER = CONPOW
      VPOWER = CON10 * ALOG10(VPOWER) + 60.
      KSENS = IRDOAT(51) * KON10
      J = IRODAT(25)
      VGAIN = ( J/2 + (J - (J/2) * 2) ) * 3 - KSENS
      UMOOE = IROOAT(38)
      UMOOE = MODE(IBIT(UMOOE,4))
      UPOWER = RDDAT(32) * CONUP * CONOFP
      IF (UPOWER .LE. 0.) UPOWER = CONPOW
      UPOWER = CON10 * ALOG10(UPOWER) + 60.
      J = IRODAT(35)
      UGAIN = ( J/2 + (J - (J/2) * 2) ) * 3 - KSENS
      KSENS = - KSENS + 10
      TKREF = IROOAT(61)
      TKREF = TKREF1(IBIT(TKREF,3))
      ITKMOO(1) = IRDDAT(62)
      ITKMOD(1) = MODTK1(ITKMOD(1))
      LOSRAN = IRDOAT(73) + 1
      LOSRAN = LOS(LOSRAN)
      ITKMOO(2) = IRDDAT(69)
      ITKMOO(2) = MODTK2(IBIT(ITKMOO(2),4))
      LOSANG = IRDOAT(74) + 1
      LOSANG = LOS(LOSANG)
      B3 = IRDDAT(85)
      B4 = IRODAT(29)
      J = IRDOAT(28)
      B5 = 1.
      IF(J.EQ.0) B5 = 10.
      IF (J .EQ. 2) B5 = 4.5
      IF (J .EQ. 4) B5 = 10.
      Y = IROOAT(22)
      IF ( Y .GE. 8.) GO TO 80
      VLEFT = PLMINX
      VRITE = PLMINX
      GO TO 81
   80 Y = SQRT(Y)
      RGC =0.
      J = IRDOAT(25)
      IF ((J - (J/2) * 2) .GT. 0) RGC = 3.
      RGC = RGC + (J/2) * 3
      TGC = IRDDAT(26) * 3
      VRITE = (B1VHF * (B3 ** 2) * (2 ** ((TGC+RGC)/6)) / (B5 * Y)) ** 2
      VLEFT = VRITE * ( XB2 **2)
      IF (VLEFT .GT. 0.) GO TO 803
      VLEFT = PLMINX
      GO TO 805
 803  VLEFT = 10. * ALOG10(VLEFT) + KSENS
 805  VRITE = VRITE * (B4 ** 2)
      IF (VRITE .GT. 0.) GO TO 807
      VRITE = PLMINX
      GO TO 808
 807  VRITE = 10. * ALOG10(VRITE) + KSENS
 808  B2 = IRDDAT(33)
      84 = IRDDAT(39)
      J = IRDDAT(38)
      B5 = 1.
      IF (J .EQ. 2) B5 = 5.01
      IF (J .EQ. 4) B5 = 11.21
      Y = IRDDAT(32)
      IF (Y .GE. 8.) GO TO 809
      ULEFT = PLMINX
      URITE = PLMINX
      GO TO 81
 809  Y = SQRT(Y)
      RGC = 0.
      J = IRDDAT(35)
      IF ((J - (J/2) *2) .GT. 0) RGC = 3.
      RGC = RGC + (J/2) * 3
      TGC = IRDDAT(36) * 3
      URITE = (B1UHF * (B3 ** 2) * (2 ** ((TGC+RGC)/6)) / (B5 * Y)) ** 2
      ULEFT = URITE * ( 82 ** 2)
      IF (ULEFT .GT. 0.) GO TO 810
      ULEFT = PLMINX
      GO TO 811
 810  ULEFT = 10. * ALOG10(ULEFT) + KSENS
 811  URITE = URITE * (B4 ** 2)
      IF (URITE .GT. 0.) GO TO 812
      URITE = PLMINX
      GO TO 81
 812  URITE = 10. * ALOG10(URITE) + KSENS
C
C SAVE DATA TO 8E PLOTTED
C
  81  IF (IDELX .LT. 0) GO TO 94
      NPLOT = NPLOT + 1
      PLOTIM(NPLOT) = TAL
      IF (VLEFT .LT. PLMAXX) GO TO 82
      XLEFT(NPLOT) = PLMAXX
      GO TO 86
  82  IF (VLEFT .GE. PLMINX) GO TO 84
      XLEFT(NPLOT) = PLMINX
      GO TO 86
  84  XLEFT(NPLOT) = VLEFT
  86  IF (VRITE .LT. PLMAXX) GO TO 88
      XRITE(NPLOT) = PLMAXX
      GO TO 881
  88  IF (VRITE .GE. PLMINX) GO TO 90
      XRITE(NPLOT) = PLMINX
      GO TO 881
 90   XRITE(NPLOT) = VRITE
881   IF (ULEFT .LT. PLMAXX) GO TO 882
      YLEFT(NPLOT) = PLMAXX
      GO TO 884
882   IF (ULEFT .GE. PLMINX) GO TO 883
      YLEFT(NPLOT) = PLMINX
      GO TO 884
883   YLEFT(NPLOT) = ULEFT
884   IF (URITE .LT. PLMAXX) GO TO 885
      YRITE(NPLOT) = PLMAXX
      GO TO 92
 885  IF (URITE .GE. PLMINX) GO TO 886
      YRITE(NPLOT) = PLMINX
      GO TO 92
 886  YRITE(NPLOT) = URITE
  92  IF (TAL .LE. TIMMAX) GO TO 94
      NUMBER = NPLOT - 1
      IF (NUMBER .EQ. 0) GO TO 1445
      GO TO 142
  94  TIME = TAL + DELTAT - .001
      IF (NUMLIN .LT. MAXLIN) GO TO 96
  95  IPAGE = IPAGE + 1
      WRITE(IOUT,6110) MESAGE,IPAGE
      WRITE(IOUT,6010)
      NUMLIN = 1
      GO TO 100
  96  NUMLIN = NUMLIN + 1
 100  IF (NUMREC .EQ. 0) GO TO 105
      NUMLIN = NUMLIN + 5
      IF (NUMLIN .GT. MAXLIN) GO TO 95
      WRITE(IOUT,6050) NUMREC
      NUMREC = 0
 105  WRITE(IOUT,6020) IGMTH,IGMTM,GMTS,TAL,ALT,RANGE,RANRAT,AZ,
     1 [illegible]ZOFF, ELOFF, VLEFT, ULEFT, VMODE, VPOWER, VGAIN,
     2IVP[illegible]NUMBUF),UMODE,UPOWER,UGAIN,ITKMCD(1),
     3 LOSRAN, ITKMOD(2), LOSANG, GATE ,TKREF,DELTRA,IRTBIT,ITLBIT
 110  K25MIL = K25MIL + 1
      IF (K25MIL .EQ. N25MIL) GO TO 115
      IRDRM = IRDRM + I25MIL
      GO TO 120
 115  K25MIL = 0
      IRDRD = IRDRD + I50MIL
      IRDRM = IRDRD + IREL
 120  CONTINUE
C
 130  IF (MFLAG .EQ. 1) GO TO 150
 140  IF (NFLAG .EQ. 1) GO TO 40
      IF (IDELX .LT. 0) GO TO 146
C
C RASTER CO-ORDINATE OF TIME = ((TIME(I) - TIMMIN) / (TIMMAX - TIMMIN))
C * (RASMAX - RASMIN) + RASMIN
C
 142  TIMDIF = TIMMAX - TIMMIN
```

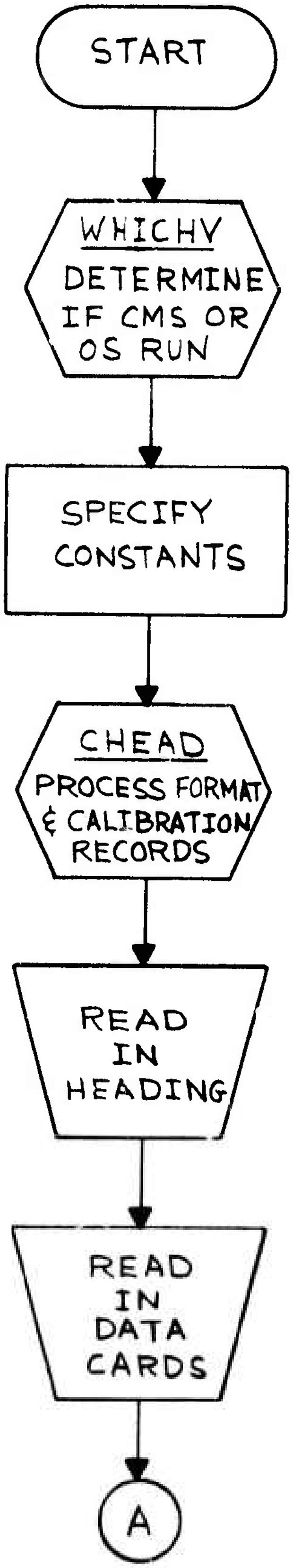
```
      DO 1425 K = 1, NUMBER
      ITIME(K) = ((PLOTIM(K) - TIMMIN) / TIMDIF) * 1860 + 1020
 1425 CONTINUE
C
      DO 144 J = 1, 4
C DRAW, TICK, AND LABEL AXES
C
      CALL METAZ (9. 2, IAX, JAX, KAX, LAX)
      CALL METAZ (9,7,KXTICK,ITICK,KXTICK,JTICK)
      CALL METAZ (9,9,IXTICK,KTICK,JXTICK,KTICK)
      DO 143 K = 1, 16
      CALL METAZ (5, 0, LOCX(K), LOCY(K), 4, LABEL(K))
 143  CONTINUE
      CALL METAZ (5, 0, IXPOS(1), IYPOS(1), 3, LABXAX)
      CALL METAZ (50, 6, 1, 0, 0, 0)
      CALL METAZ (5, 0, IXPOS(2), IYPOS(2), 18, LABYAX(1,J))
      CALL METAZ (50, 6, 0, 0, 0, 0)
C
C RASTER CO-ORDINATE OF CROSS SECTION = ((X(I) - PLMINX) / (PLMAXX -
C PLMINX)) * (RASMAX - RASMIN) + RASMIN
C
      DO 1435 K = 1, NUMBER
      JCROSS(K,J) = ((CROSEC(K,J) - PLMINX) / RCSDIF) * 2480 + 400
 1435 CONTINUE
C
C DRAW CURVE AND ADVANCE FRAME
C
      CALL METAZ (8, NUMBER, ITIME(1), JCROSS(1,J), 0,0)
      CALL METAZ (13, 1, 0, 0, 0, 0)
 144  CONTINUE
      GO TO (1443,145),NFLAG
C
C INITIALIZE NEXT PAIR OF FRAMES
C
 1443 XLEFT(1) = XLEFT(NPLOT)
      XRITE(1) = XRITE(NPLOT)
      YLEFT(1) = YLEFT(NPLOT)
      YRITE(1) = YRITE(NPLOT)
      PLOTIM(1) = PLOTIM(NPLOT)
      NPLOT = 1
 1445 LABEL(1) = TIMMAX
      DO 1447 K = 2, 7
      LABEL(K) = LABEL(K-1) + IDELT
 1447 CONTINUE
      TIMMIN = LABEL(1)
      TIMMAX = LABEL(7)
      IF (PLOTIM(1) .LE. TIMMAX) GO TO 640
 630  IF (PLOTIM(1) .LE. TIMMAX + DELSIX) GO TO 1445
      TIMMAX = TIMMAX + DELSIX
      GO TO 630
 640  WRITE (99, 6090) (LABEL(K), K = 1, 7)
      READ (99, 6080) (LABEL(K), K = 1, 7)
      GO TO 94
145   CALL PLTND
C
 146  IF (NUMBUF .GT. 1) GO TO 330
 147  IF(IOUT.EQ.8) CALL EXIT
```

```
      RETURN
C
 150  IF (TAL2(NUMBUF) .LT. TSTART) GO TC 40
      IF (TAL2(NUMBUF) .LE. TSTOP) GO TO 160
      MFLAG = 2
      GD TD 140
C
 160  NCHECK = 4
      ITAR2(NUMBUF) = -1
      NTGRN = ICADAT(11) - 2
      IF (NTGRN .NE. 0) GD TD 162
      NTGRD = 0
      NVHF2(NUMBUF) = 0
      NUHF2(NUMBUF) = C
      NVHFD = 0
      NUHFD = 0
      GD TD 290
C
 162  NVHF2(NUMBUF) = ICACAT(21) - 1
      NUHF2(NUMBUF) = NTGRN - NVHF2(NUMBUF)
      IDENT = IACC + ICADAT(17) * 3
      NTCTE = ICADAT(8)
      DO 180 I = 1,NTDTE
      IDBASE = IDENT + (I - 1) * 3
      K = IDBDAT(1) + 1
      IF(IDBCAT(2).EO.0) GD TD 180
      NVTRG = ICADAT(21)
      IAMB(NTOTE) = IDBDAT(3)
      JT = 2
      IF(IAMB(NTOTE).LT.NVTRG) JT = 1
 180  CCNTINUE
 182  NTGRD = NTGRN
      NVHFD = NVHF2(NUMBUF)
      NUHFC = NUHF2(NUMBUF)
 184  IRDRM = IRDRC2 + IREL
      IF (IRDRM .EC. IREL) GD TO 290
      J = ICADAT(1C)
      IF (J .EC. 0) GD TD 290
C
C GET TIMES AND RANGES FROM RDRM
C
      I50MIL = ICADAT(13) * 3
      NRDRM = ICADAT(12) * N25MIL
      K25MIL = 0
      DO 200 I = 1, NRCRM
      CALL RADART (FMRDRM, IRDRM, IGMTH, IGMTM, GMTS, TRTAL2(I))
      RANGE2(I) = (RDDAT(85) + RDDAT(86)) * CCNRAN
      K25MIL = K25MIL + 1
      IF (K25MIL .EQ. N25MIL) GO TD 190
      IRDRM = IRDRM + I25MIL
 190  K25MIL = 0
      IRDRD2 = IRCRD2 + I50MIL
      IRDRM = IRCRC2 + IREL
 200  CCNTINUE
C
C DETERMINE THE BEST MATCH BETWEEN MINOR CYCLE TIMES AND 25 MS DATA
C BLCCK TIMES. SAVE THE NUMBER OF THAT MINOR CYCLE AND THE RANGE FROM
C THAT 25 MS DATA BLOCK.
```

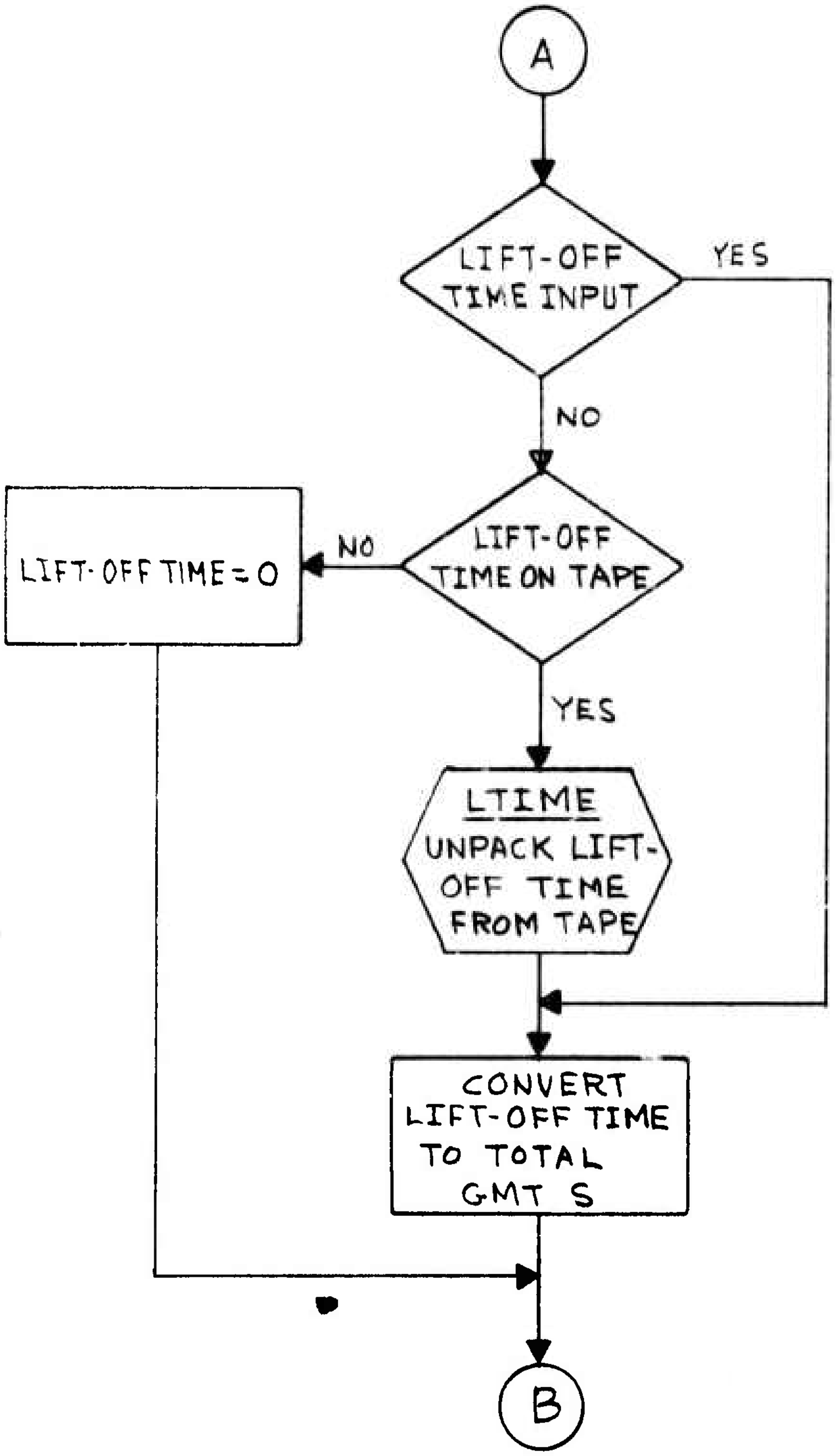
```
C
      DTIME2 = 1000000.
      DELTIM = (ICADAT(5) / 1000000.) * ICADAT(7)
      DO 210 I = 1, J
      TMINOR = TAL2(NUMBUF) + (I - 1) * DELTIM
      DO 210 K = 1, NRCRM
      DELTA = DABS(TMINOR - TRTAL2(K))
      IF (DELTA .GE. DTIME2) GO TO 210
      DTIME2 = DELTA
      MINCR = I
      TRANGE = RANGE2(K)
 210  CONTINUE
C
C GET THE RANGES FOR ALL TARGETS FROM THE MINCR CYCLE DETERMINED ABOVE.
C DETERMINE WHICH RANGE IS CLOSEST TO THE ONE CHOSEN ABOVE AND CALL
C THAT THE TRACKED TARGET.
C
      ITRBAS = IADD + (ICADAT(18) + (MINOR - 1) * (NTGRN +2) * 3) * 3
      K = 0
      DELRAN = 1000000.
      IF (IDENT .NE. IADD) GO TO 220
      GO TO (290, 215), KFLAG
 215  IFLAG = 2
      GO TO 230
 220  IFLAG = 1
      KFLAG = 2
      NTOT = NTOTE
 230  IF (NTOT .EQ. 0) GO TO 290
      DO 270 I = 1, NTOT
      GO TO (250, 240), IFLAG
 240  K = K + 1
      GO TO 260
 250  IDBASE = IDENT + (I - 1) * 3
      J = IDBDAT(2)
      IF (J .EQ. 0) GO TO 270
      K = K + 1
      IAMB(K) = IDBDAT(3) * 3
      ITARG(K) = IDBDAT(1)
      IF (ITARG(K) .GT. 1) GO TO 255
      K = K - 1
      GO TO 270
 255  IPRI(K) = IDBDAT(4)
 260  J = ITRBAS + IAMB(K)
      TDBRAN = IGET(FMCTDB, J, 1) * .0018737031
      DELTA = ABS(TRANGE - TDBRAN)
      IF (DELTA .GE. DELRAN) GO TO 270
      DELRAN = DELTA
      ITAR2(NUMBUF) = ITARG(K)
      J = ITAR2(NUMBUF) + 1
      ITYPE2(NUMBUF) = JTYPE(JT)
      IPRIO2(NUMBUF) = IPRI(K)
      DRANG2(NUMBUF) = DELTA
      NCHECK = 7
 270  CONTINUE
      NTOT = K
 290  DO 300 I = 1, NCHECK
      IF ( IBUFF(NUMBUF,I) .NE. IBUFO(I) ) GO TO 310
```

```
  300 CCNTINUE
      GO TO 40
  310 DO 320 I = 1, NCHECK
      IBUFO(I) = IBUFF(NUMBUF,I)
  320 CCNTINUE
      TAGTI2(NUMBUF) = (CATDAT(1) * I2T22 + CATDAT(2)) / KCNTAG
      NUMBUF = NUMBUF + 1
      IF (NUMBUF .LE. MAXBUF) GO TO 40
  330 J = 1
  340 IPAGE = IPAGE + 1
      WRITE(IOUT,6110) MESAGE,IPAGE
      WRITE(IOUT,6030)
      DO 370 I = 1, MAXLI2
      IF (ITAR2(J) .NE. -1) GO TO 350
      WRITE(IOUT,6040) IGMTH2(J),IGMTM2(J),GMTS2(J),
     1TAL2(J), TAGTI2(J), (IBUFF(J,K), K = 1, 4)
      GO TO 360
  350 WRITE(IOUT,6040) IGMTH2(J),IGMTM2(J),GMTS2(J),
     1TAL2(J), TAGTI2(J), (IBUFF(J,K), K = 1, 7), DRANG2(J)
 360  IF (J .GE. NUMBUF - 1) GO TO 380
      J = J + 1
  370 CCNTINUE
      GO TO 340
C
  380 IF ( (NFLAG .EQ. 2) .AND. (MFLAG .EQ. 2) ) GO TO 390
      NUMLIN = MAXLIN
      NUMBUF = 1
      GO TO 40
   390 IF(IOUT.EQ.8) CALL EXIT
      RETURN
      END
```

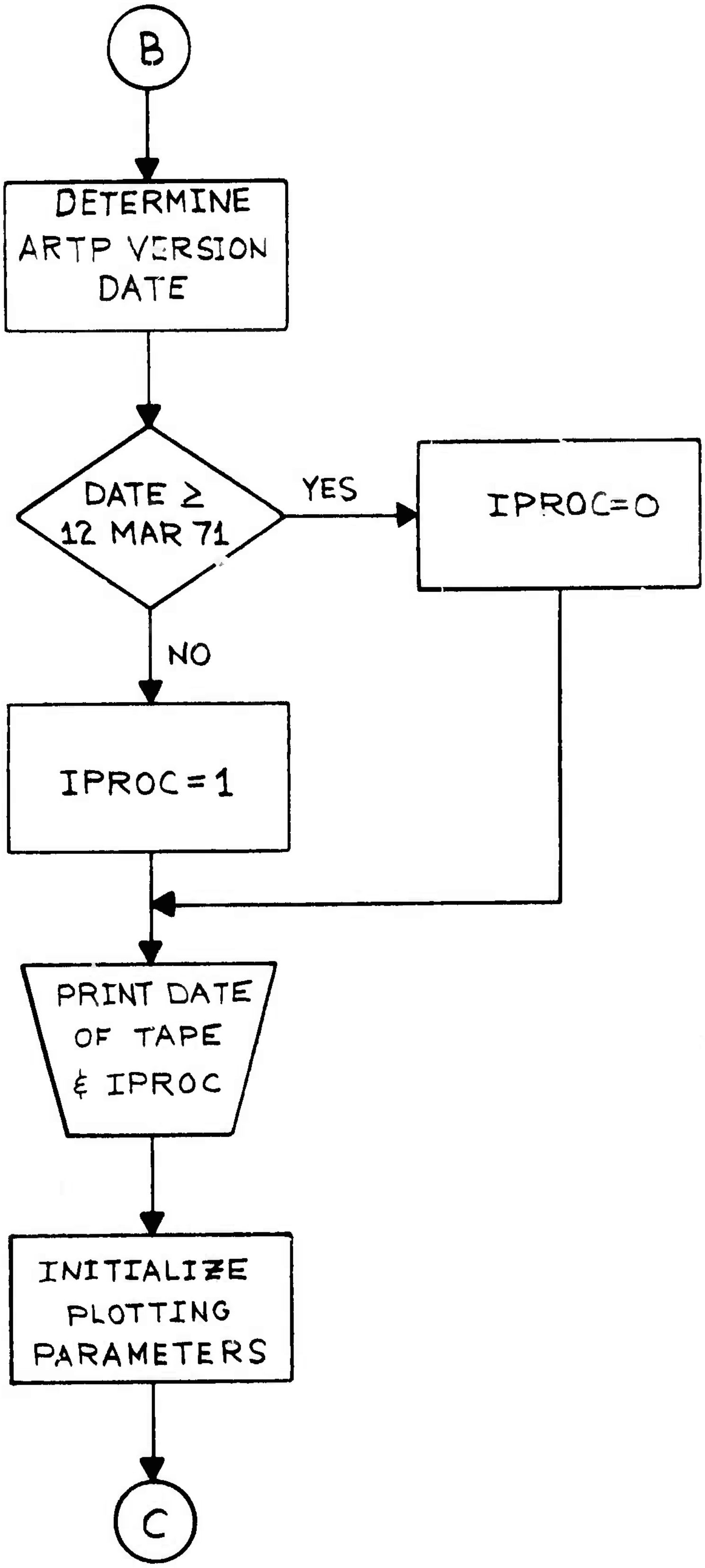
APPENDIX D
ALTCEP FLOW DIAGRAM



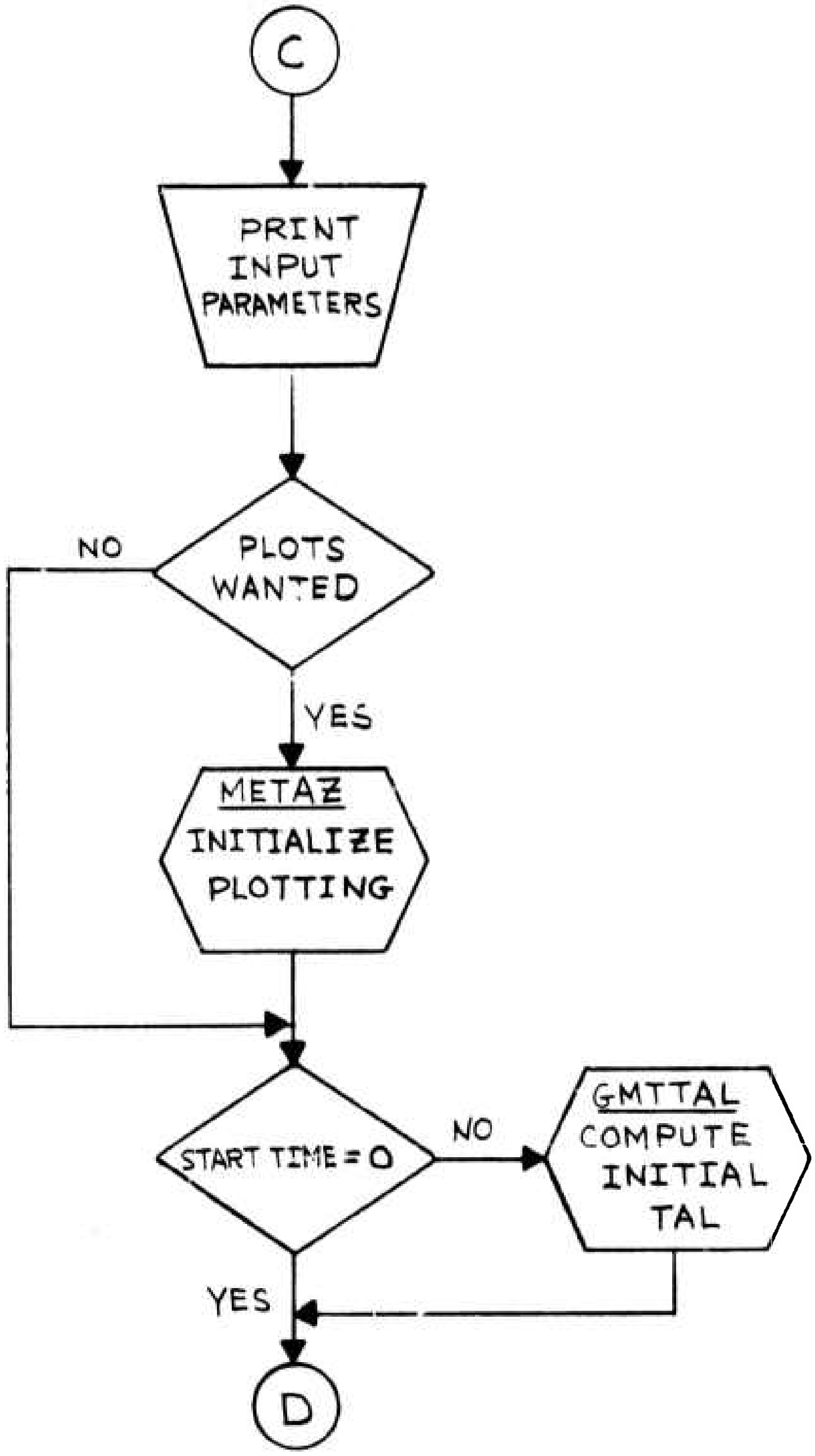
APPENDIX D-2


A
LIFT-OFF TIME INPUT
YES
NO
LIFT-OFF TIME ON TAPE
NO
LIFT-OFF TIME = 0
YES
LTIME
UNPACK LIFT-OFF TIME FROM TAPE
CONVERT LIFT-OFF TIME TO TOTAL GMT S
B

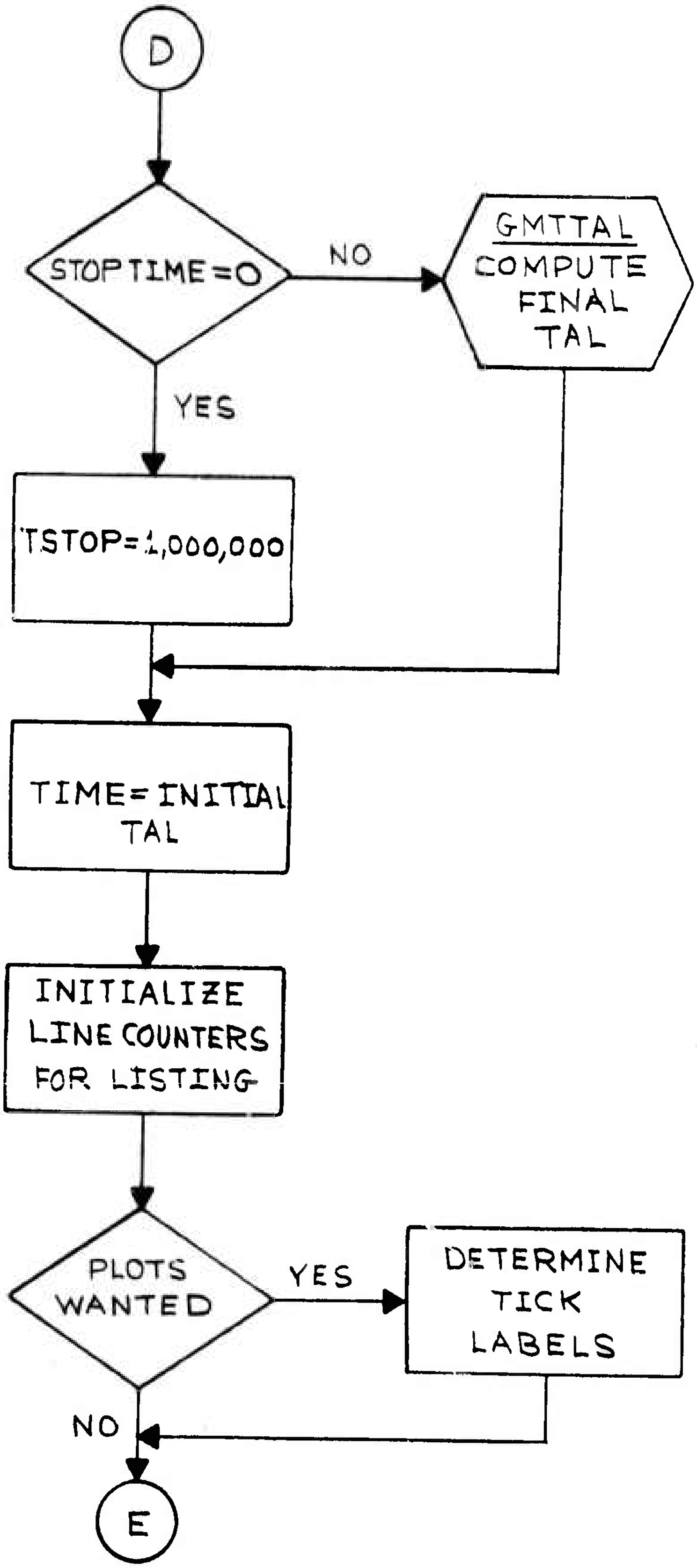
APPENDIX D-3


B
DETERMINE ARTP VERSION DATE
DATE ≥ 12 MAR 71
YES
IPROC=0
NO
IPROC=1
PRINT DATE OF TAPE & IPROC
INITIALIZE PLOTTING PARAMETERS
C

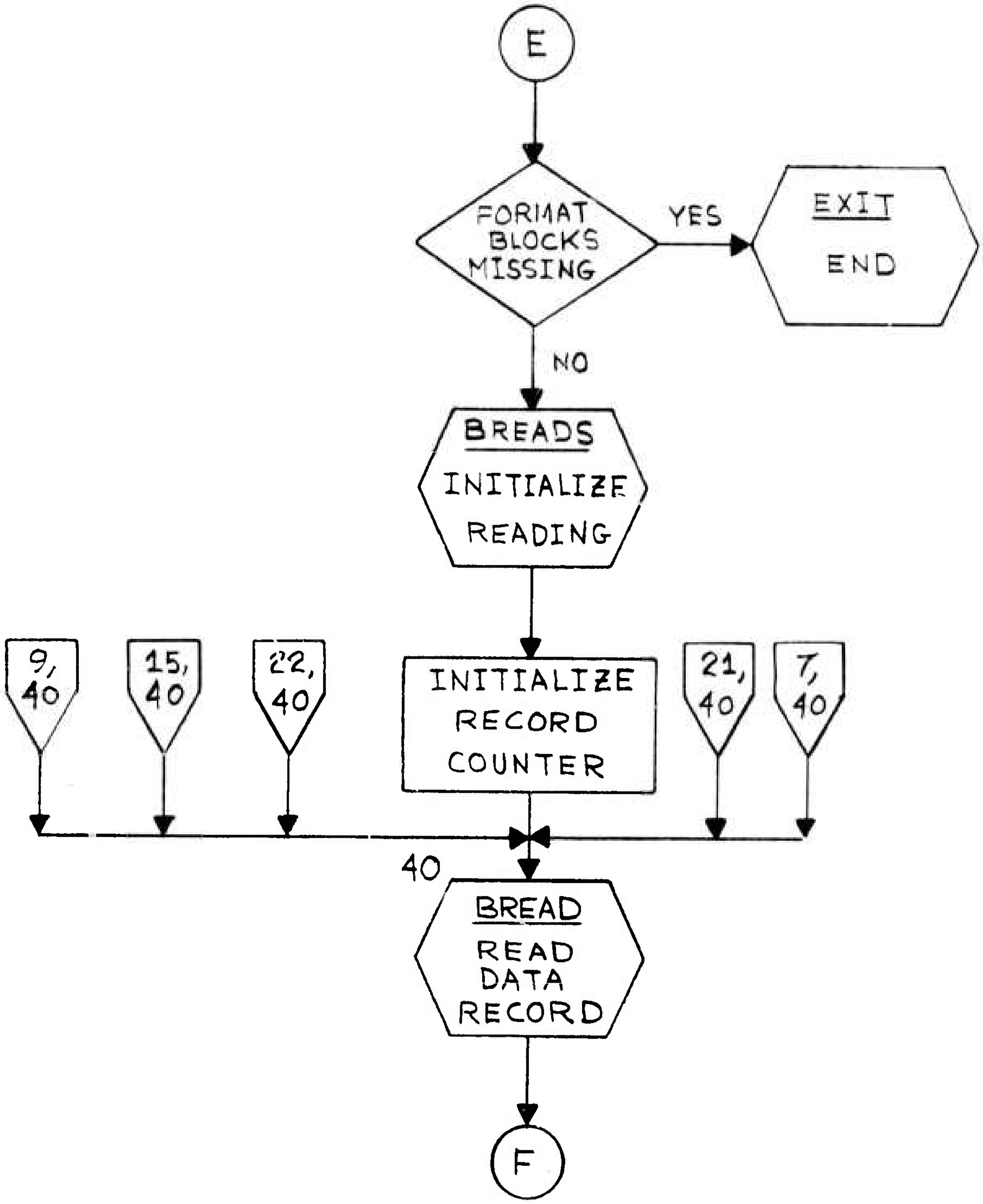
APPENDIX D-4


C
PRINT INPUT PARAMETERS
PLOTS WANTED
NO
YES
METAZ
INITIALIZE PLOTTING
START TIME = 0
NO
GMTTAL
COMPUTE INITIAL TAL
YES
D

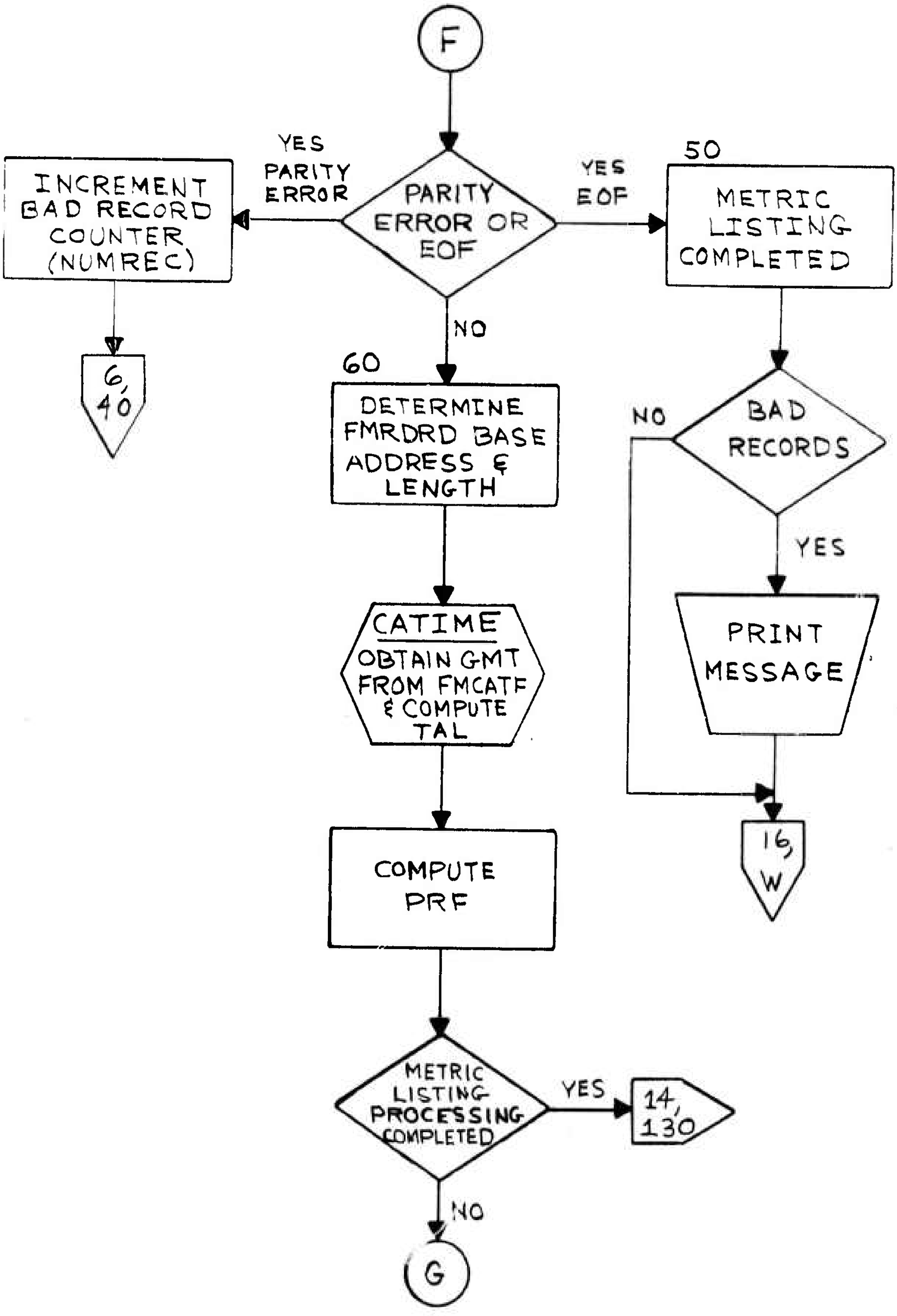
APPENDIX D-5


D
STOPTIME=0
NO
GMTTAL
COMPUTE
FINAL
TAL
YES
TSTOP=1,000,000
TIME=INITIAL
TAL
INITIALIZE
LINE COUNTERS
FOR LISTING
PLOTS
WANTED
YES
DETERMINE
TICK
LABELS
NO
E

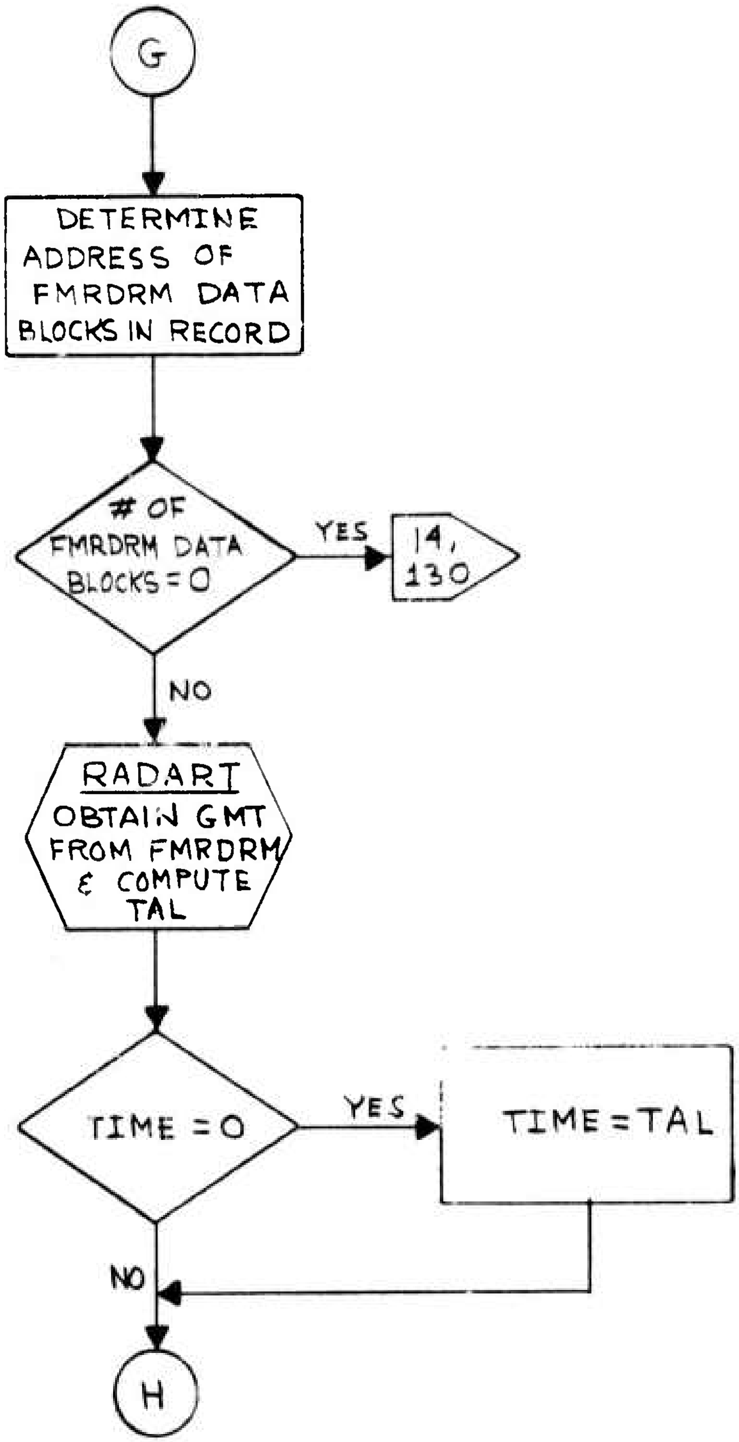
APPENDIX D-6


E
FORMAT BLOCKS MISSING
YES
EXIT
END
NO
BREADS
INITIALIZE READING
9, 40
15, 40
22, 40
INITIALIZE RECORD COUNTER
21, 40
7, 40
40
BREAD
READ DATA RECORD
F

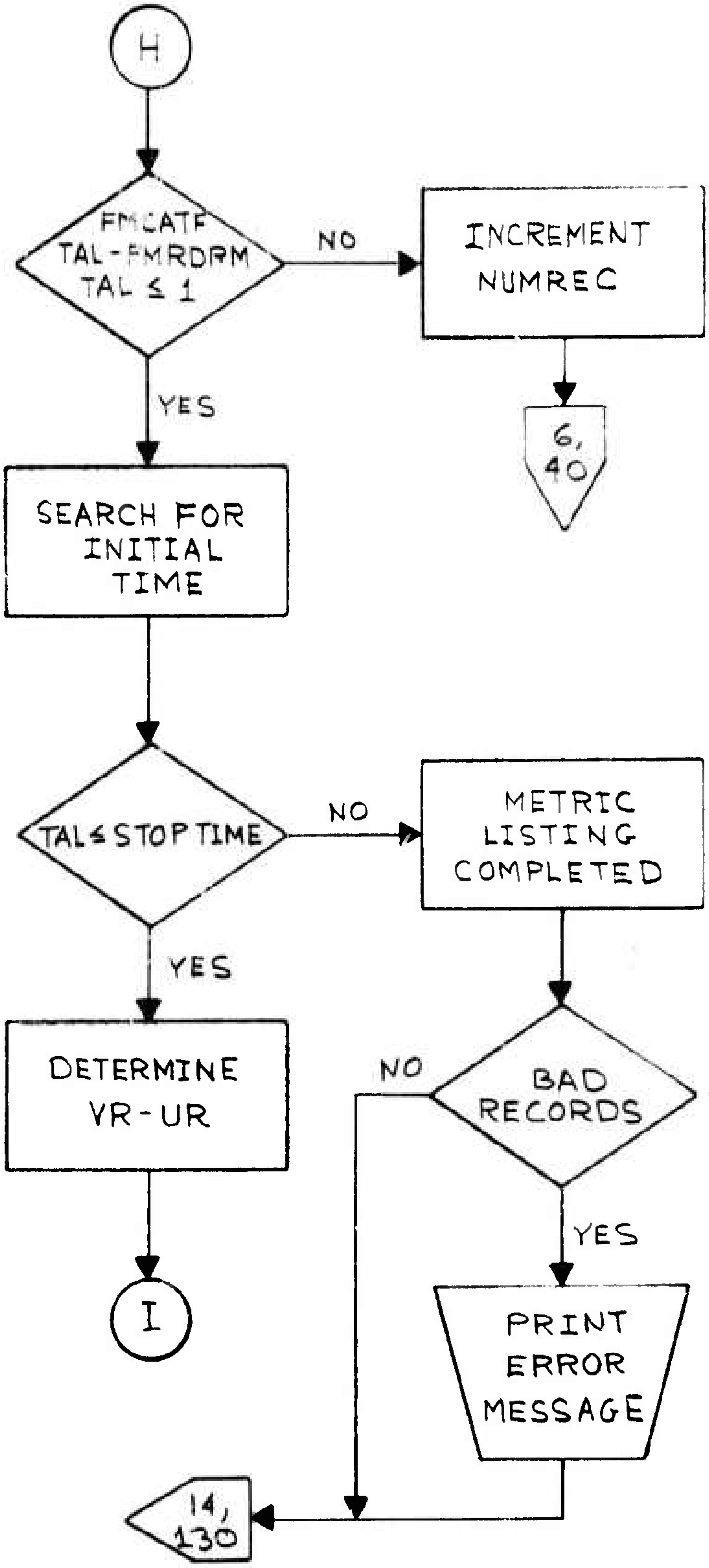
APPENDIX D-7


F
YES
PARITY
ERROR
PARITY
ERROR OR
EOF
YES
EOF
INCREMENT
BAD RECORD
COUNTER
(NUMREC)
50
METRIC
LISTING
COMPLETED
6,
40
NO
60
DETERMINE
FMRDRD BASE
ADDRESS &
LENGTH
NO
BAD
RECORDS
YES
CATIME
OBTAIN GMT
FROM FMCATF
& COMPUTE
TAL
PRINT
MESSAGE
COMPUTE
PRF
16,
W
METRIC
LISTING
PROCESSING
COMPLETED
YES
14,
130
NO
G

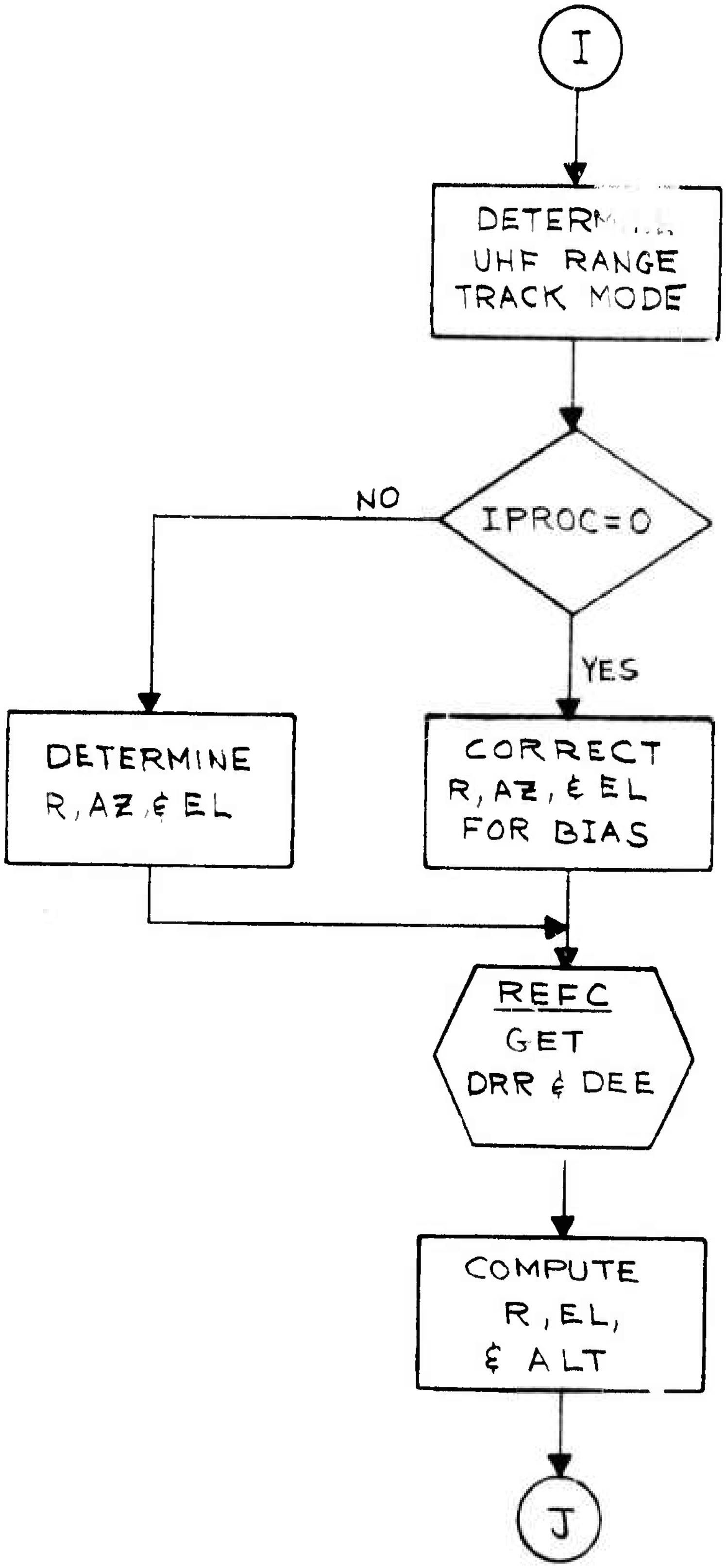
APPENDIX D-8


G
DETERMINE ADDRESS OF FMRDRM DATA BLOCKS IN RECORD
# OF FMRDRM DATA BLOCKS = O
YES
14, 130
NO
RADART
OBTAIN GMT FROM FMRDRM & COMPUTE TAL
TIME = O
YES
TIME = TAL
NO
H

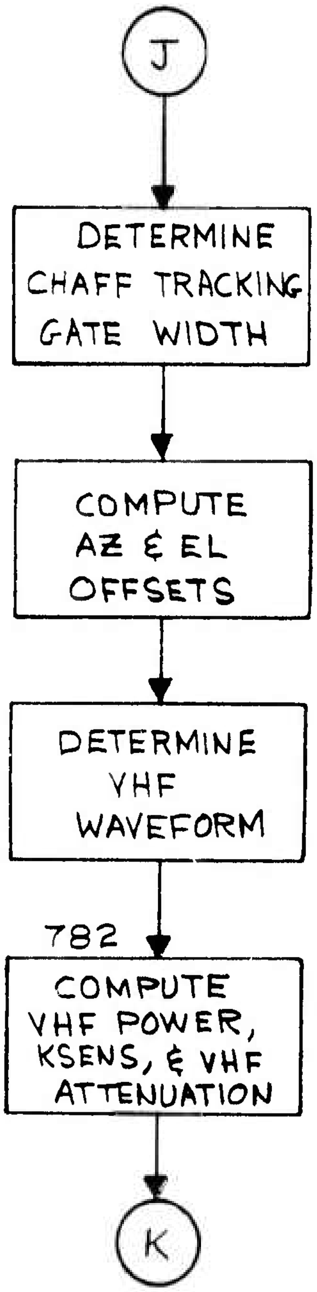
APPENDIX D-9


H
FMCATF
TAL-FMRDRM
TAL ≤ 1
NO
INCREMENT
NUMREC
YES
6,
40
SEARCH FOR
INITIAL
TIME
TAL≤STOP TIME
NO
METRIC
LISTING
COMPLETED
YES
DETERMINE
VR-UR
NO
BAD
RECORDS
YES
I
PRINT
ERROR
MESSAGE
14,
130

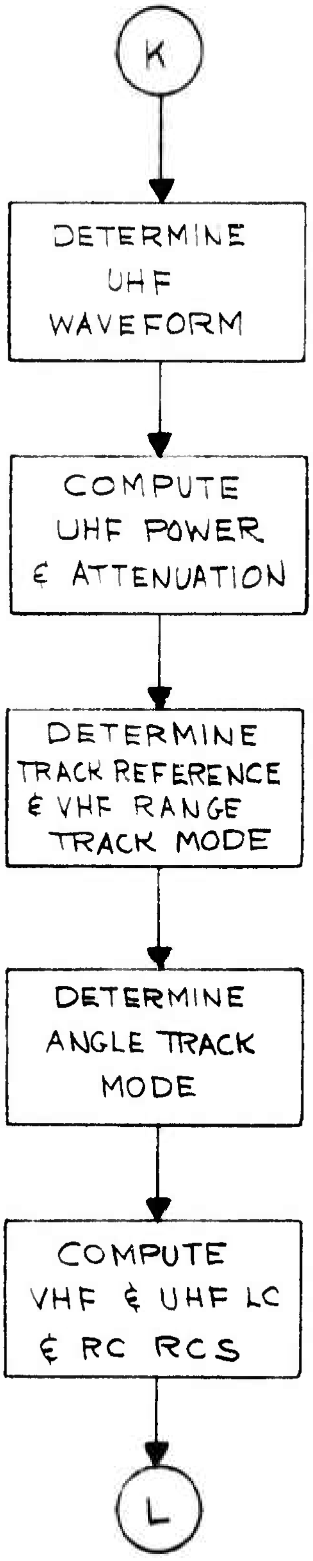
APPENDIX D-10



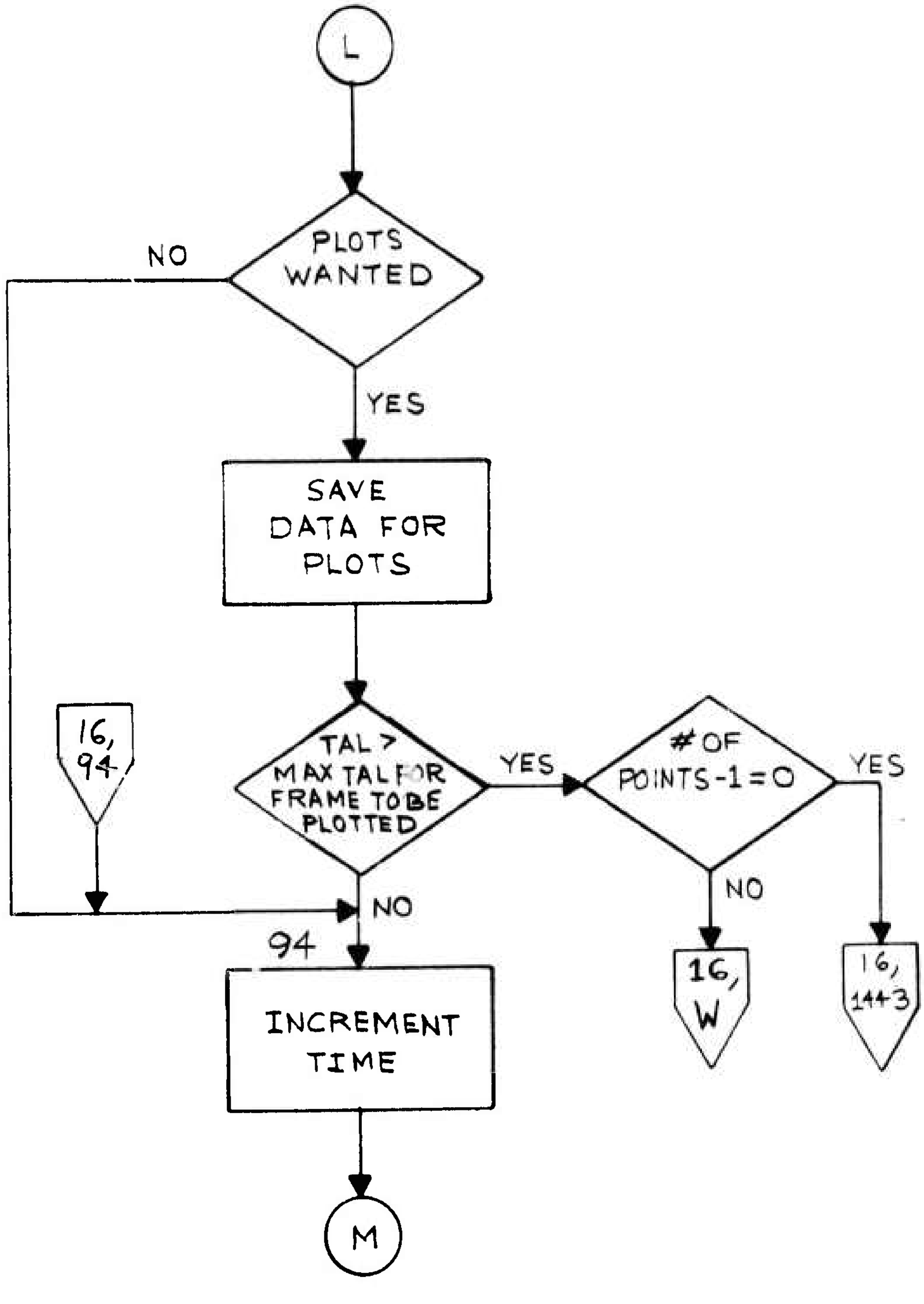
APPENDIX D-11


J
DETERMINE CHAFF TRACKING GATE WIDTH
COMPUTE AZ & EL OFFSETS
DETERMINE VHF WAVEFORM
782
COMPUTE VHF POWER, KSENS, & VHF ATTENUATION
K

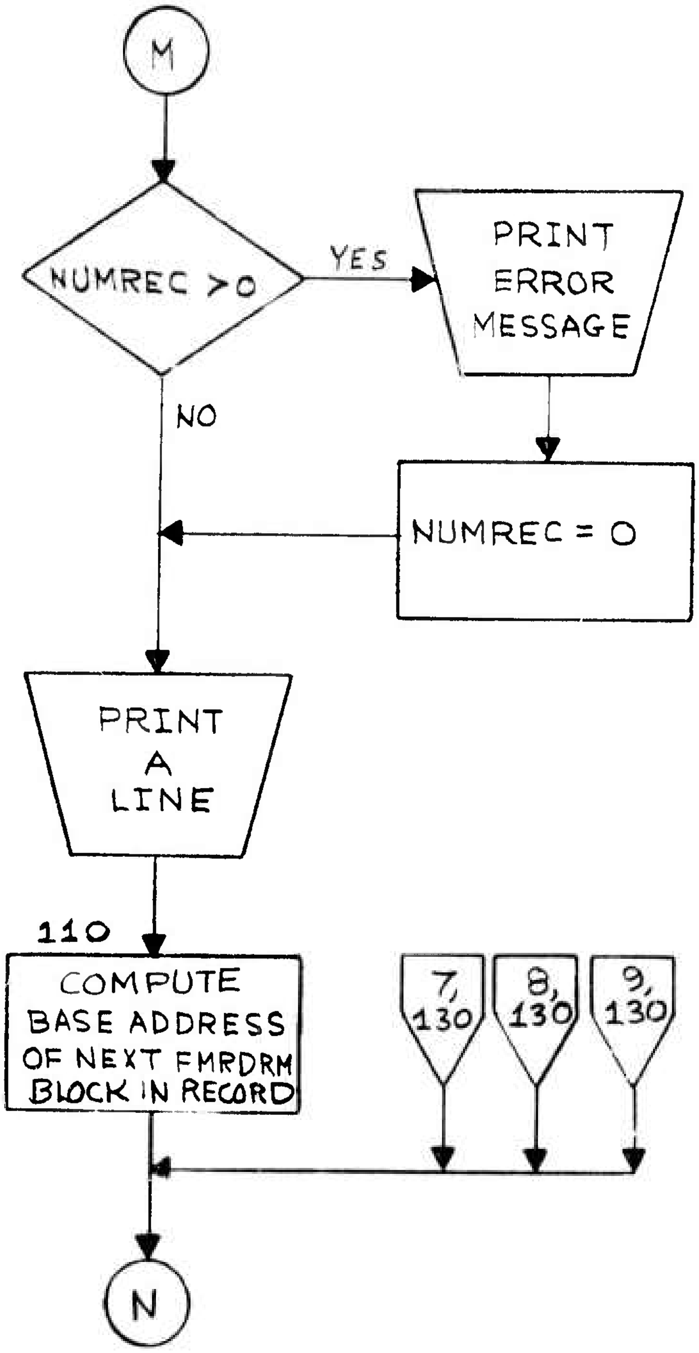
APPENDIX D-12


K
DETERMINE UHF WAVEFORM
COMPUTE UHF POWER & ATTENUATION
DETERMINE TRACK REFERENCE & VHF RANGE TRACK MODE
DETERMINE ANGLE TRACK MODE
COMPUTE VHF & UHF LC & RC RCS
L

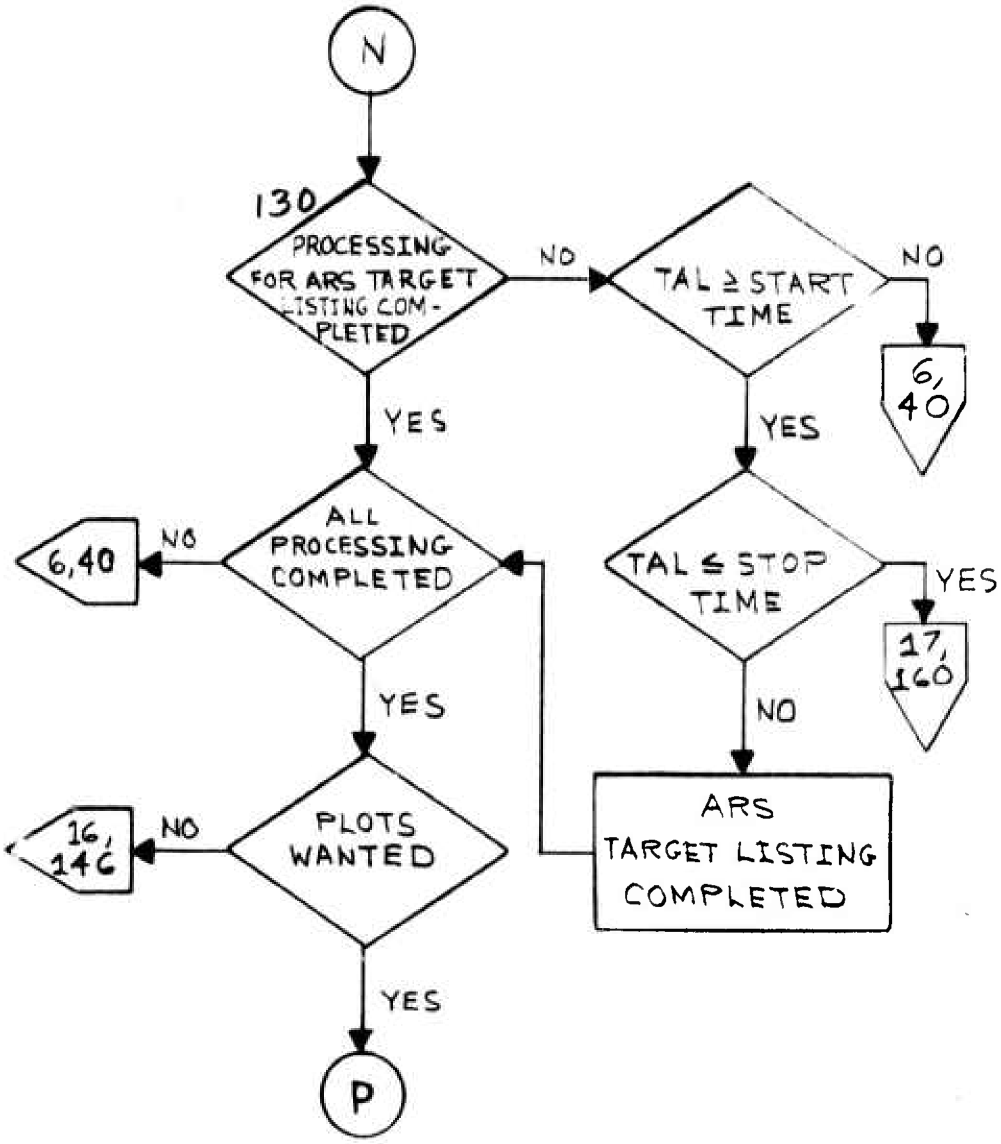
APPENDIX D-13



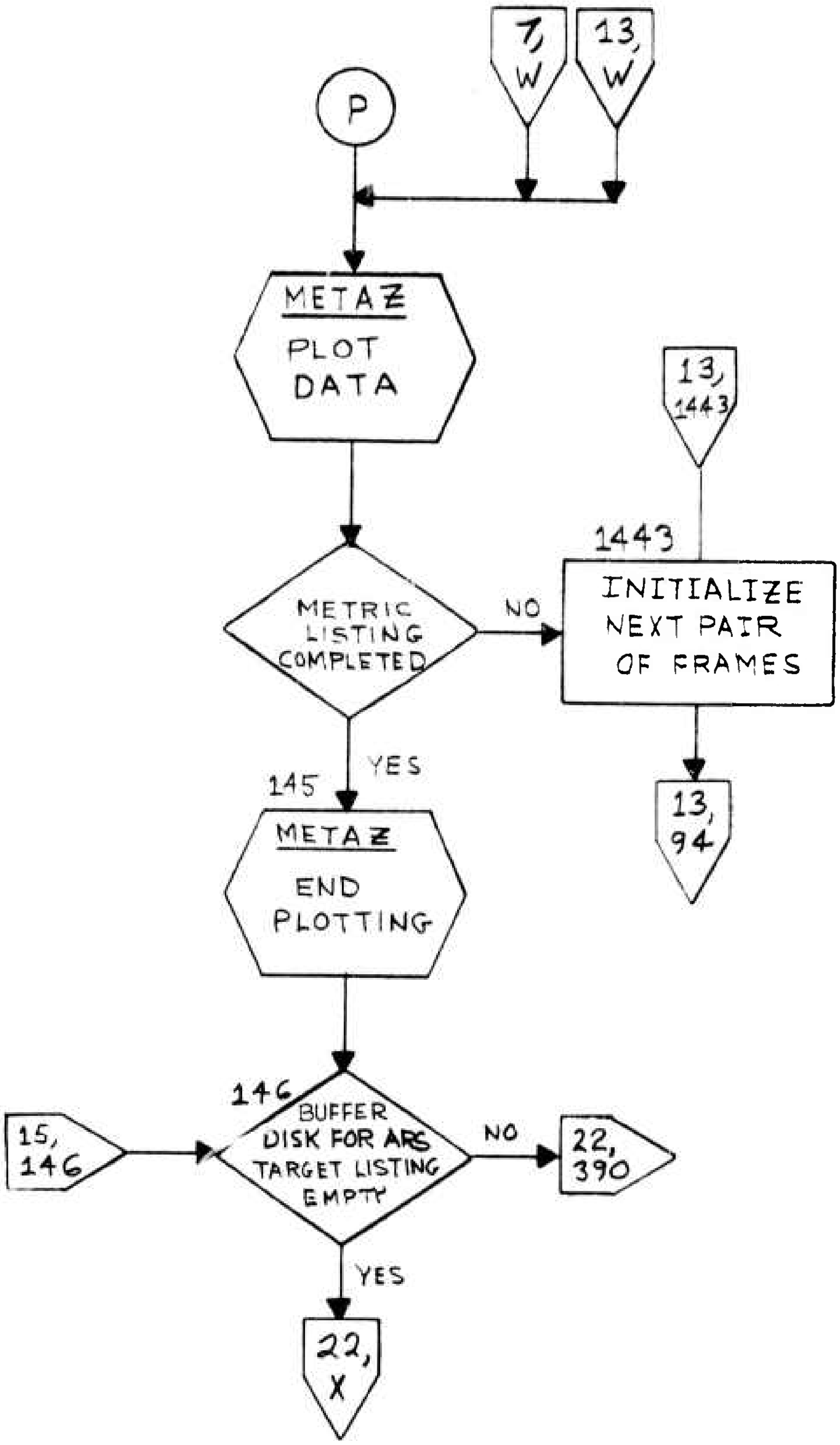
APPENDIX D-14



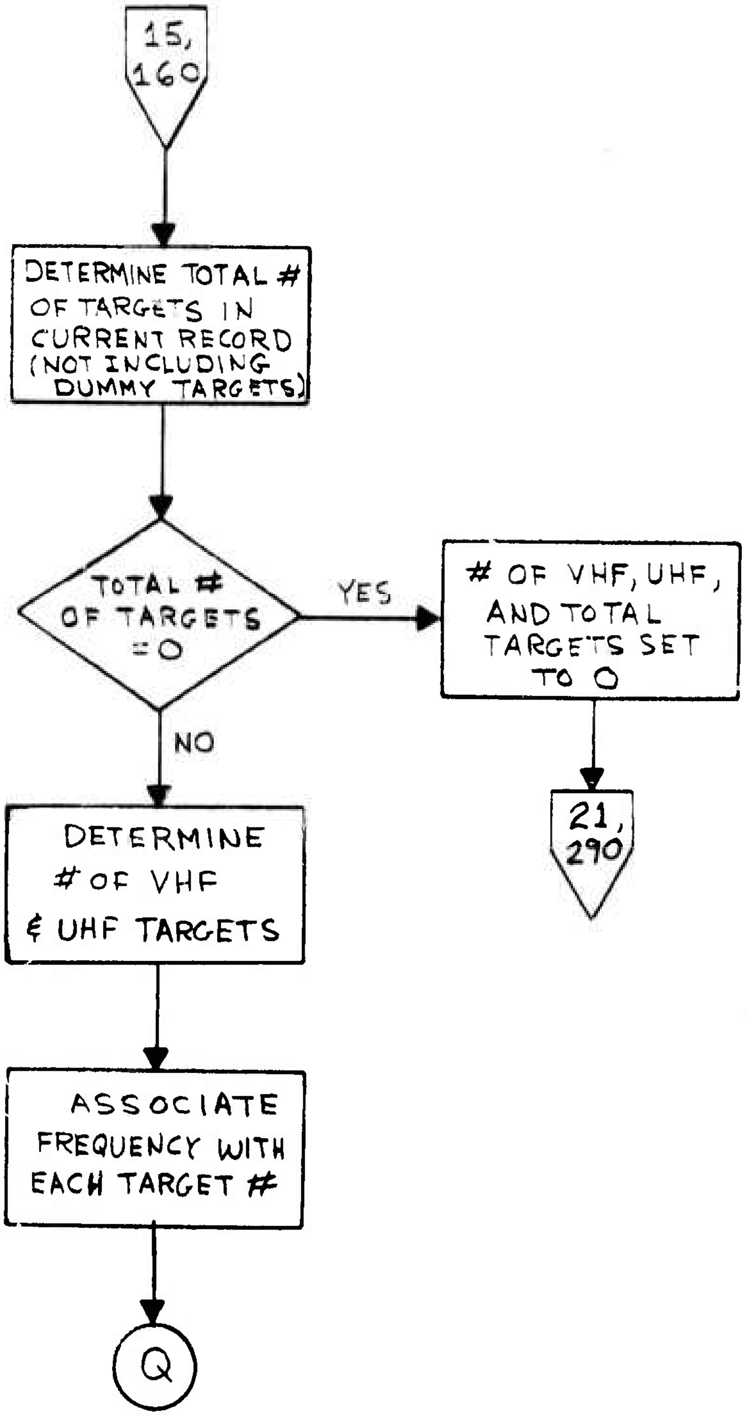
APPENDIX D-15



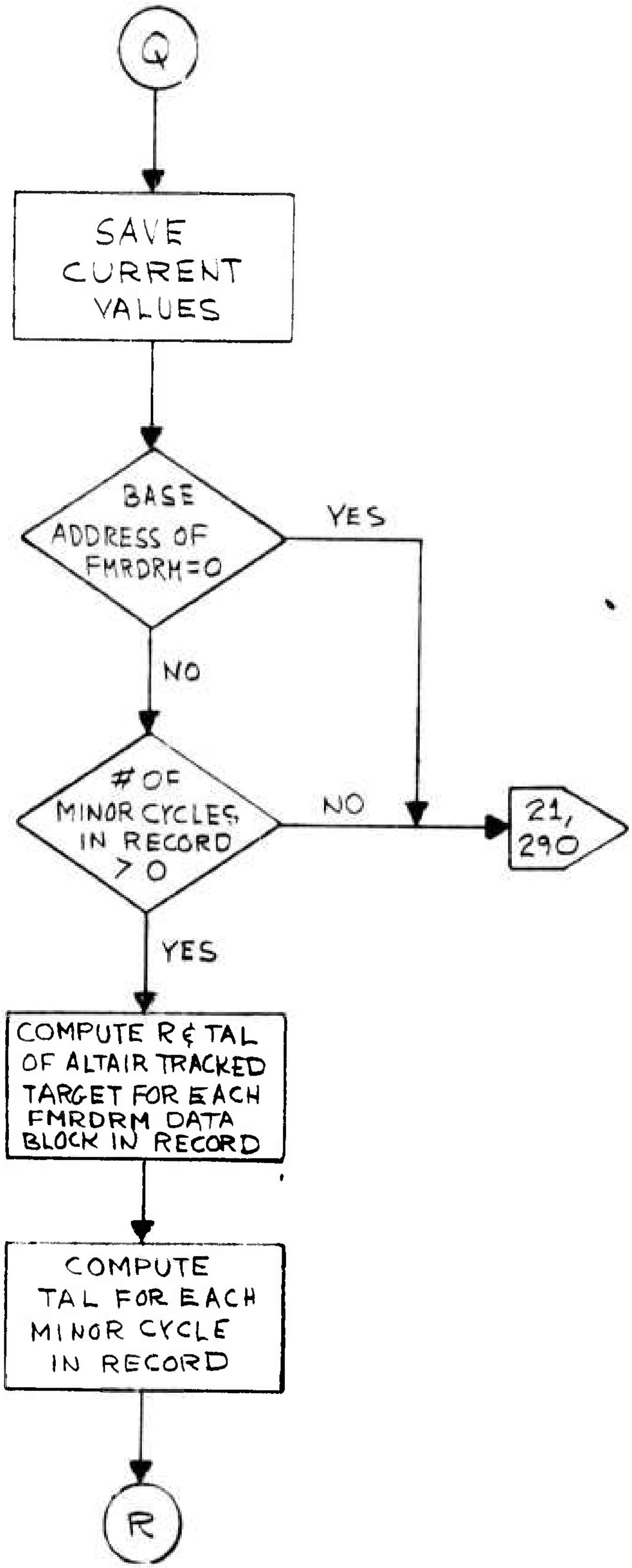
APPENDIX D-16



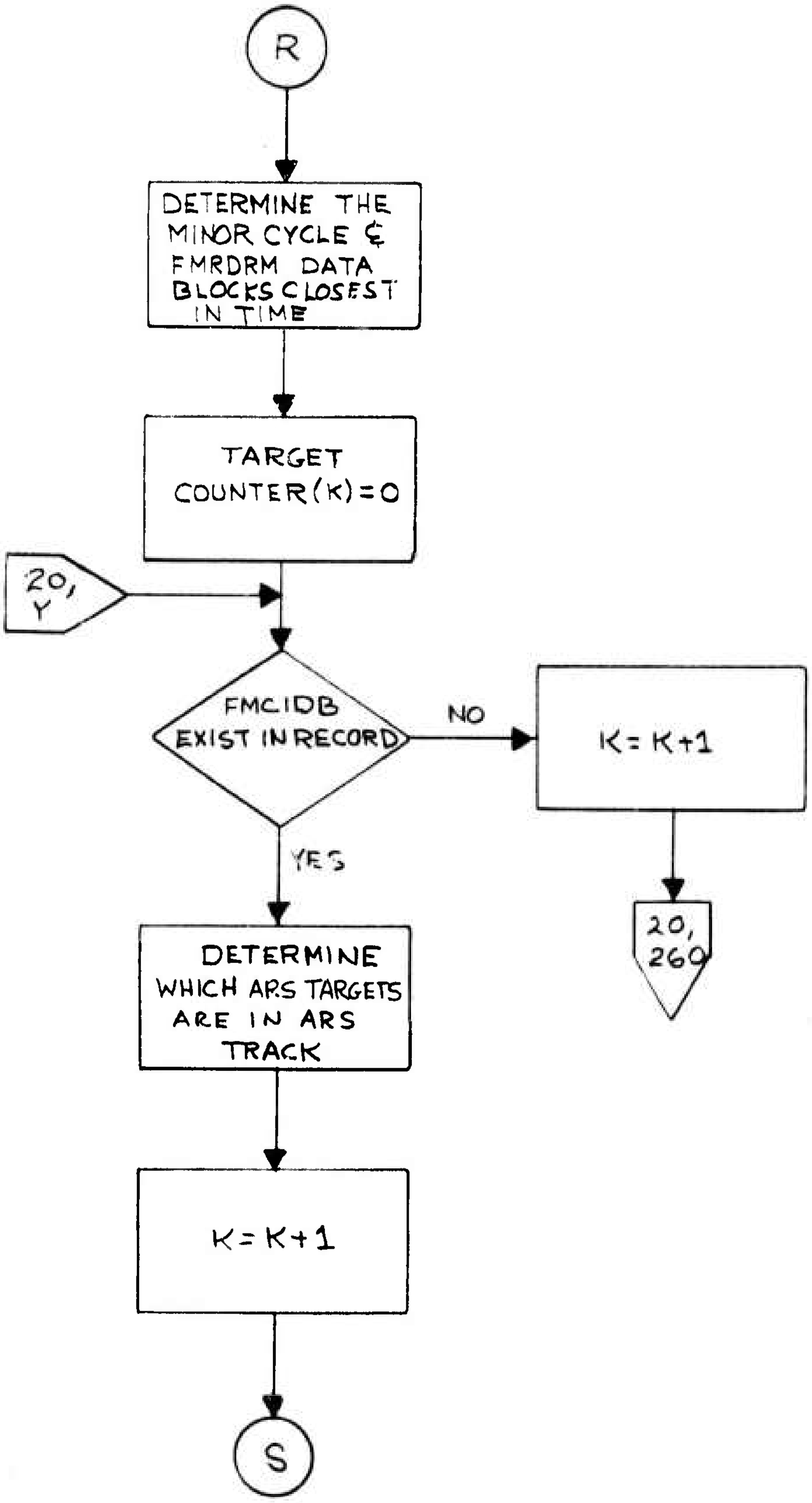
APPENDIX D-17


15,
160
DETERMINE TOTAL # OF TARGETS IN CURRENT RECORD (NOT INCLUDING DUMMY TARGETS)
TOTAL # OF TARGETS = O
YES
# OF VHF, UHF, AND TOTAL TARGETS SET TO O
21,
290
NO
DETERMINE # OF VHF & UHF TARGETS
ASSOCIATE FREQUENCY WITH EACH TARGET #
Q

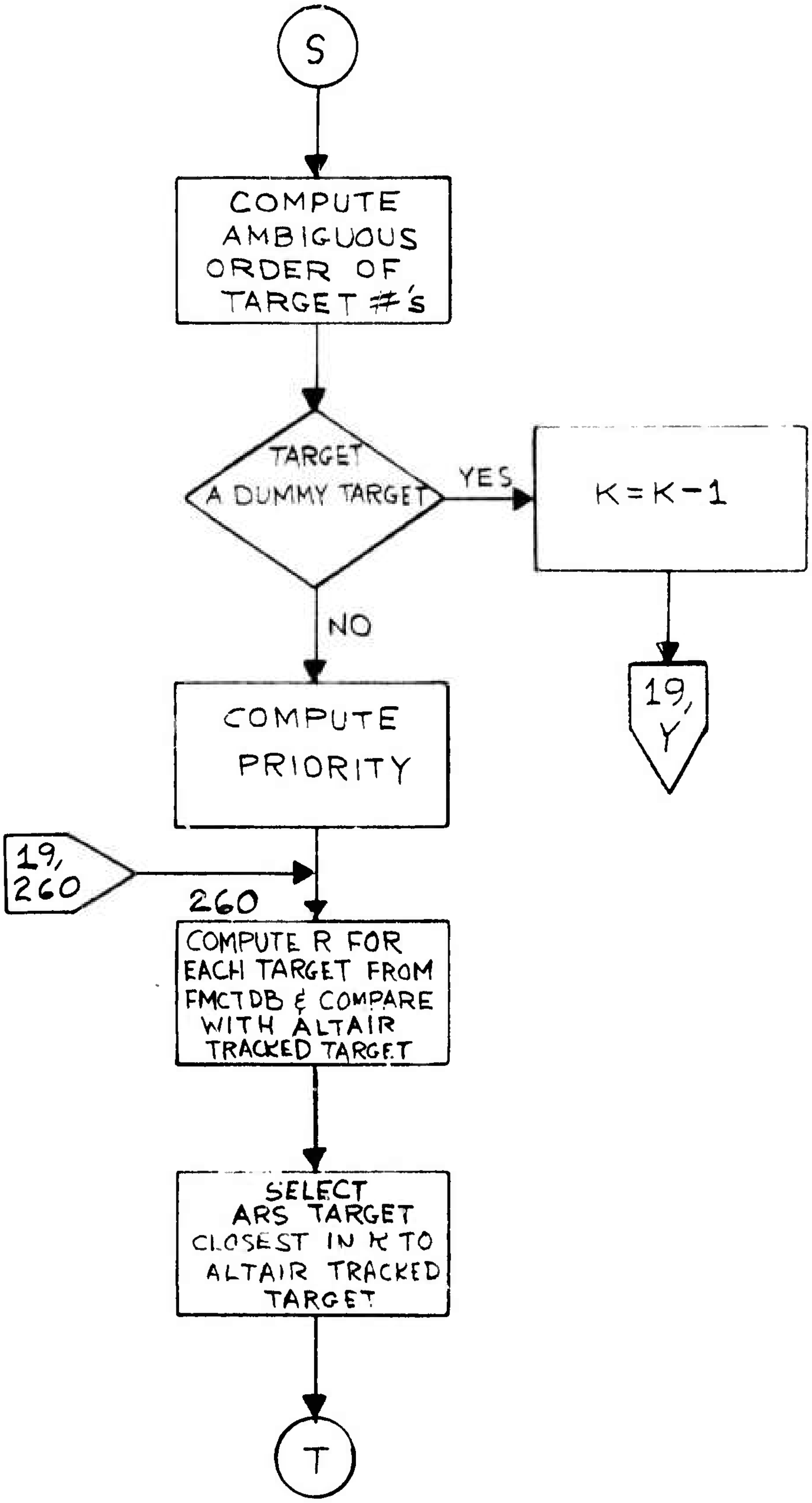
APPENDIX D-18


Q
SAVE CURRENT VALUES
BASE ADDRESS OF FMRDRM=0
YES
NO
# OF MINOR CYCLES IN RECORD > 0
NO
21, 290
YES
COMPUTE R & TAL OF ALTAIR TRACKED TARGET FOR EACH FMRDRM DATA BLOCK IN RECORD
COMPUTE TAL FOR EACH MINOR CYCLE IN RECORD
R

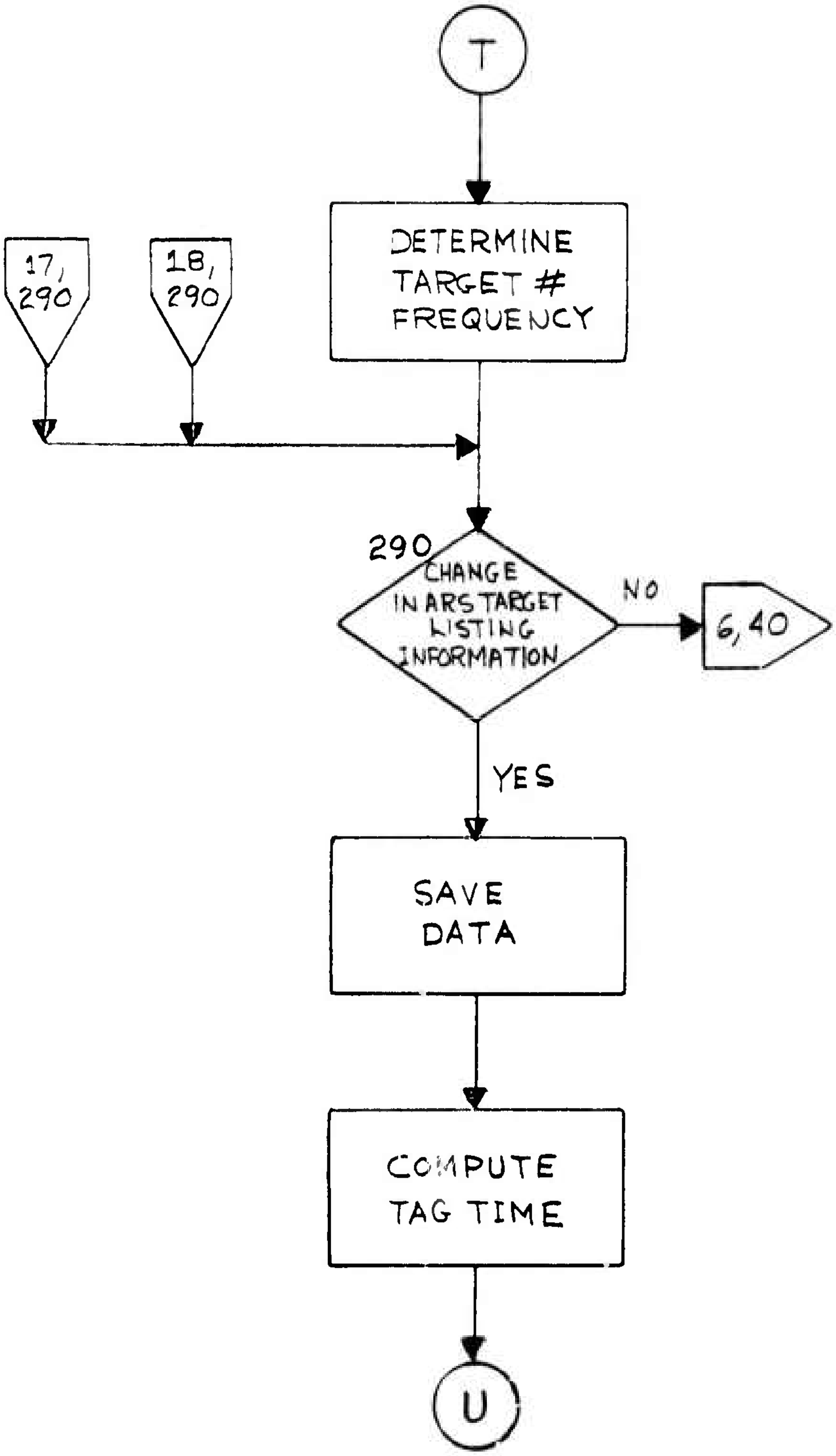
APPENDIX D-19


R
DETERMINE THE MINOR CYCLE & FMRDRM DATA BLOCKS CLOSEST IN TIME
TARGET COUNTER(K)=0
20, Y
FMCIDB EXIST IN RECORD
NO
K=K+1
20, 260
YES
DETERMINE WHICH ARS TARGETS ARE IN ARS TRACK
K=K+1
S

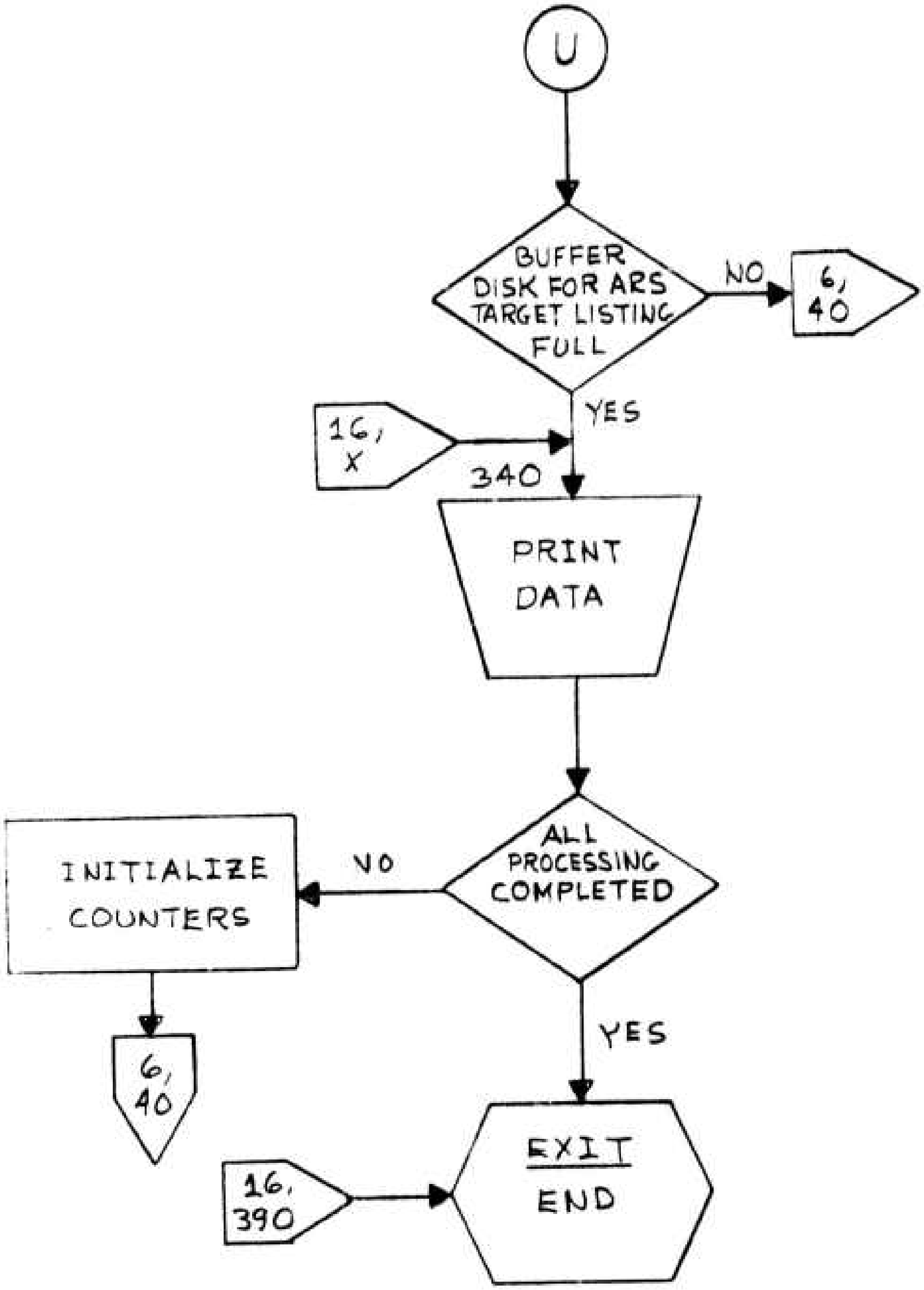
APPENDIX D-20


S
COMPUTE AMBIGUOUS ORDER OF TARGET #'s
TARGET A DUMMY TARGET
YES
K=K-1
19, Y
NO
COMPUTE PRIORITY
19, 260
260
COMPUTE R FOR EACH TARGET FROM FMCTDB & COMPARE WITH ALTAIR TRACKED TARGET
SELECT ARS TARGET CLOSEST IN R TO ALTAIR TRACKED TARGET
T

APPENDIX D-21


T
DETERMINE TARGET # FREQUENCY
17, 290
18, 290
290
CHANGE IN ARS TARGET LISTING INFORMATION
NO
6, 40
YES
SAVE DATA
COMPUTE TAG TIME
U

APPENDIX D-22

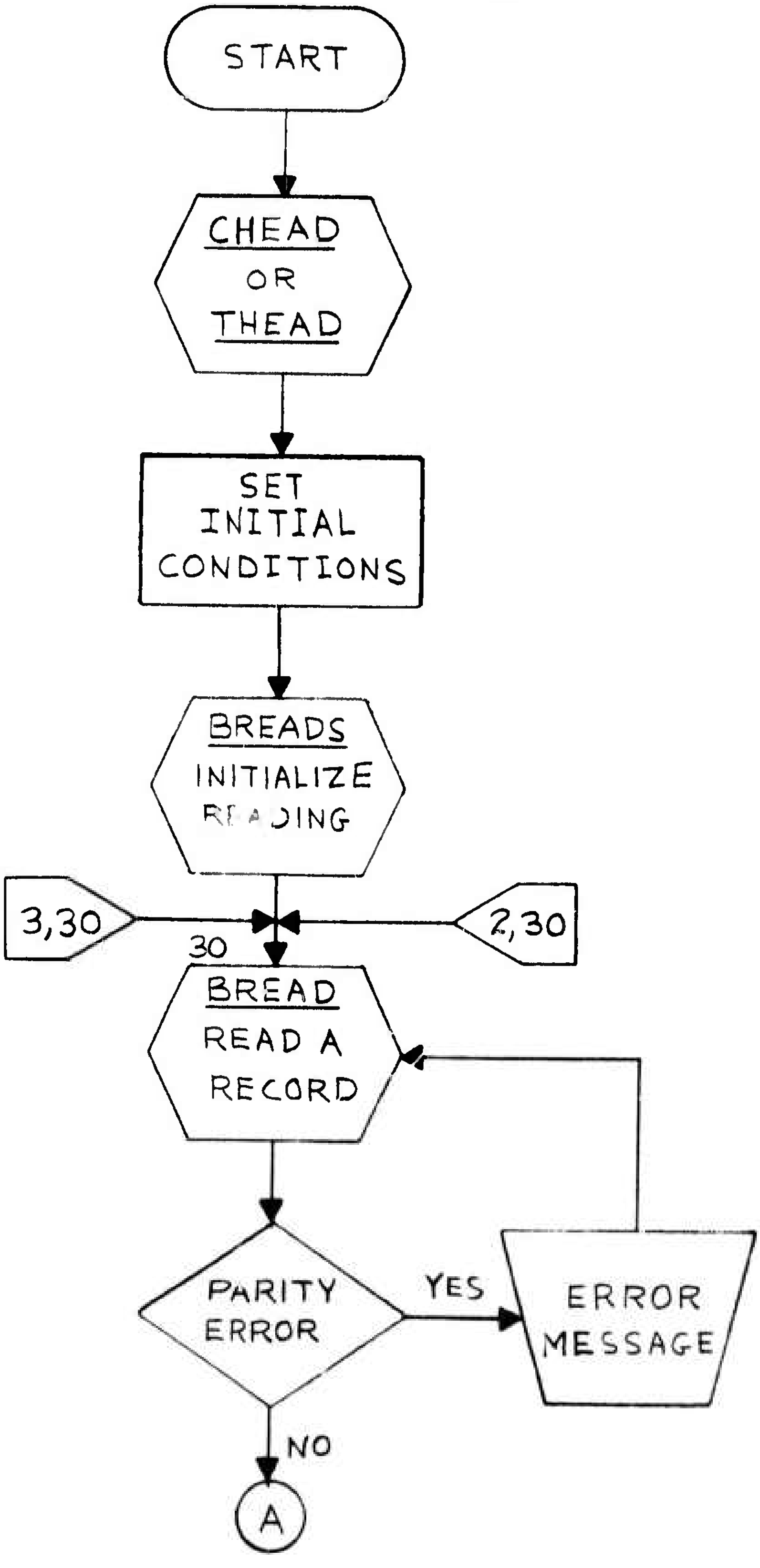


APPENDIX E
SUBROUTINE CHEAD PROGRAM LISTING

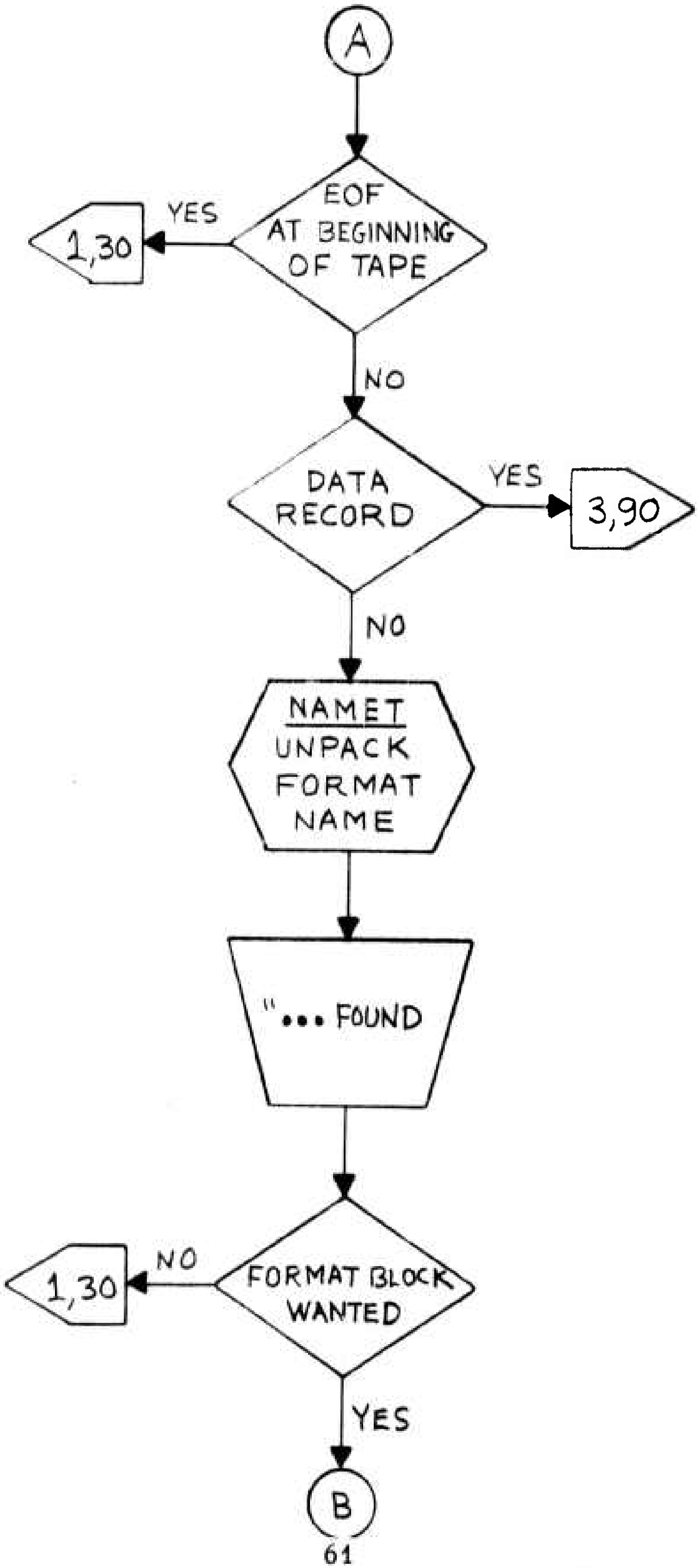
```
      SUBROUTINE CHEAD(*)
C
C     VERSION 04/01/71      R.H. FRENCH
C
C     THIS SUBROUTINE IS USED TO PROCESS ALTAIR CATALOG TAPE DATA
C     RECORDS.
C
C     IN THIS VERSION FORMAT TABLE FMRR11 HAS BEEN ADDED. THE IDARR
C     ARRAY HAS BEEN PUT IN COMMON SO THAT THE RTS AND ARTP VERSION
C     DATES CAN BE PRINTED OUT BY THE MAIN PROGRAM. THESE DATE(S) CAN
C     ALSO BE USED TO TRIGGER THE FLOW OF THE MAIN PROGRAM.
C
      INTEGER*2 ITEM
      INTEGER*2 IBU,IBU2
      COMMON/BEAD/LN,IFLG,IAOO,FMRD1D,FMCATF,FMCSAD,FMCMDB,FMCTIB,FMCIDB
     1,FMCTDB,FMRDRO,FMRDRM,FMRDRT,FMGLOT,FMRRD5,FMAXSP,FMBIAS,FMR5CH
     2,FMRCHF,FMAACC,FMRR11,NAME(19),NI(18),IX(18),ITEM(8000)
      COMMON/TITLE/IDARR(10)
      DIMENSION IBUF1(2048),IBUF2(2048),MCATF(18),NAMEX(19)
      EQUIVALENCE (FMRDIO,MCATF(1))
      EQUIVALENCE (IBUF1(1),IBU),(IBUF2(1),IBU2)
      DATA NAMEX/'ROID','CATF','CSAD','CMDB','CTIB','CIDB','CTDB','RDRD'
     1,'RDRM','RDRT','GLOT','RR05','AXSP','BIAS','R5CH','RCHF','AACC'
     2,'RR11','HDRD'/
      DATA MAX/8192/,IFL/0/,IT/1/
      NREC=0
      LN=1
      DO 20 I=1,18
      MCATF(I)=0
      NAME(I)=NAMEX(I)
      NI(I)=0
20    IX(I)=0
      NAME(19)=NAMEX(19)
      CALL BREADS(LN,IBUF1,IBUF2,MAX,IFL,INDX,LEN,IFLG,IACD)
30    CALL BREAD(LN)
      NREC=NREC+1
      IF(IFLG.EQ.2)GO TO 55
      IF(IFLG.EQ.3.AND.IT.EQ.1)GO TO 30
      IF(IFLG.EQ.3)GO TO 90
      GO TO (21,22),INDX
21    IF(IBU/256.EQ.1)GO TO 90
      GO TO 34
22    IF(IBU2/256.EQ.1)GO TO 90
34    CALL NAMET(IAOD,NAMED)
      IF(NAMED.EQ.NAMEX(19))GO TO 36
      WRITE(6,85)NAMED
85    FORMAT(1X,A4,'  FOUND')
36    DO 18 I=1,19
      IF(NAMED.EQ.NAMEX(I))GO TO 35
18    CONTINUE
      GO TO 30
35    CALL FORM(IADO,ITEM(IT),IB,NAMED,NTEM,&70)
      IF(NAMED.EQ.NAMEX(19))GO TO 66
      DO 40 I=1,18
      IF(NAME(I).EQ.NAMED)GO TO 60
40    CONTINUE
```

```
      GO TO 30
55    WRITE(6,56)NREC
56    FORMAT(' PARITY ERROR READING FORMAT RECORD',I6)
      GO TO 30
60    MCATF(I)=IB
      WRITE(6,9)NAMED,NTEM
9     FORMAT(' FORMAT=',A4,'  STORED IN COMMON  NTEM=',I4)
      IX(I)=IT
      NI(I)=NTEM
      IT=IT+6*NTEM
      GO TO 30
66    CALL HDRR(IADD,ITYP,IDARR)
      WRITE(6.72)ITYP
72    FORMAT(' TYPE ',I2)
      IF(ITYP.EQ.1)GO TO 30
      WRITE(6,68)
68    FORMAT(' TAPE NOT CATALOG TAPE JOB TERMINATED BY CHEAD')
      RETURN 1
70    WRITE(6,80)NAMED,NTEM
80    FORMAT(' NAME = 'A4,'   NTEM ='I5,' *ERROR* FORMAT TABLE LIMITED T
     10 700 ITEMS OR FORMAT TABLE HAS 0 LENGTH')
90    DO 81 I=1,18
      IF(MCATF(I).EQ.0)WRITE(6,82)NAMEX(I)
82    FORMAT(1X,A4,' NOT FOUND')
81    CONTINUE
      WRITE(6,150)
150   FORMAT(' CHEAD COMPLETE')
      RETURN
      END
```

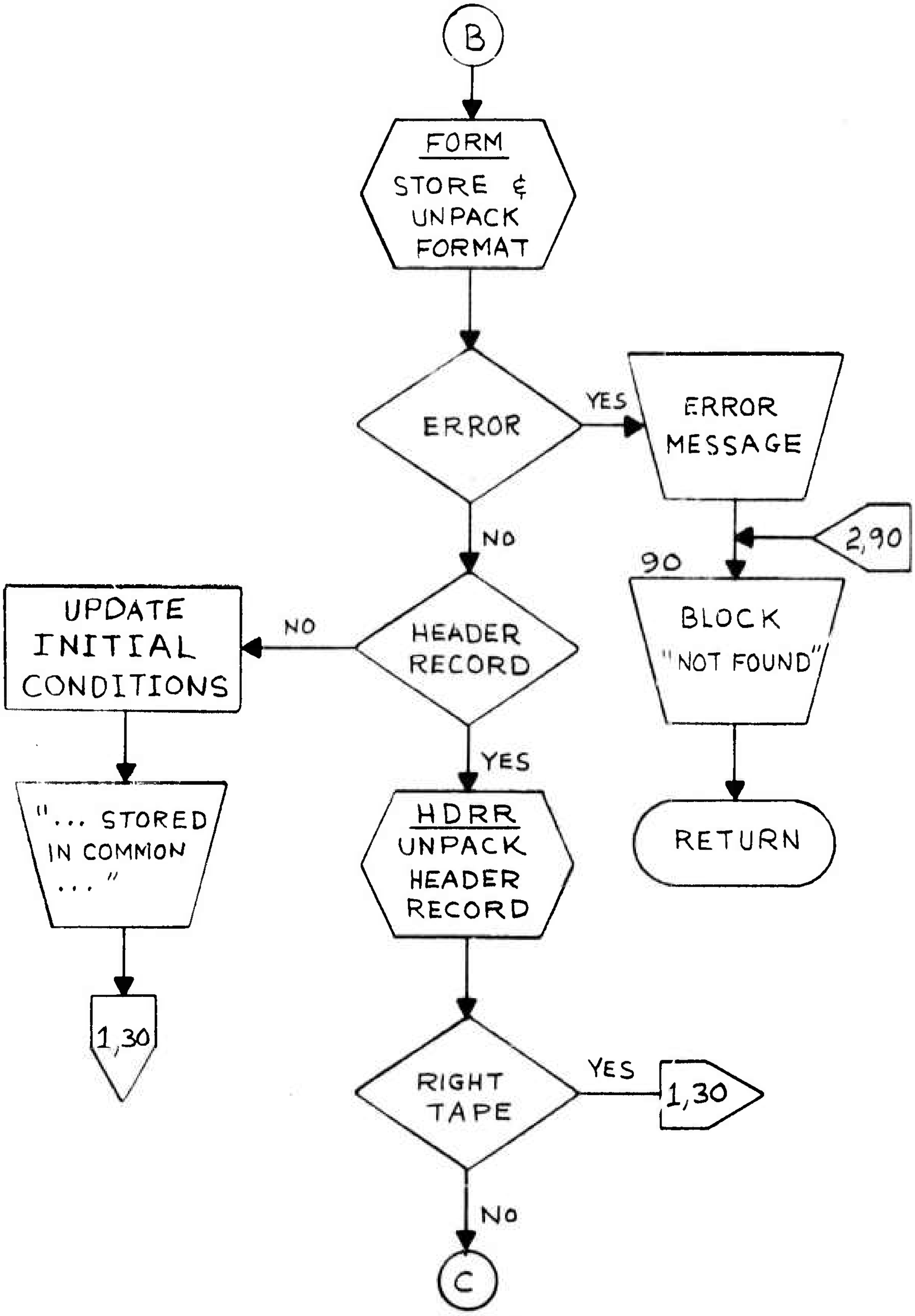
APPENDIX F
SUBROUTINE CHEAD FLOW DIAGRAM



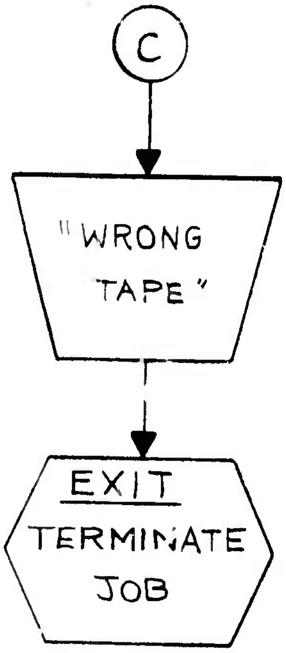
APPENDIX F-2


A
EOF AT BEGINNING OF TAPE
YES
1,30
NO
DATA RECORD
YES
3,90
NO
NAMET
UNPACK FORMAT NAME
"... FOUND
FORMAT BLOCK WANTED
NO
1,30
YES
B

APPENDIX F-3


B
FORM
STORE & UNPACK FORMAT
ERROR
YES
ERROR MESSAGE
NO
2,90
90
BLOCK "NOT FOUND"
RETURN
UPDATE INITIAL CONDITIONS
NO
HEADER RECORD
YES
"... STORED IN COMMON ..."
HDRR
UNPACK HEADER RECORD
1,30
RIGHT TAPE
YES
1,30
NO
C

APPENDIX F-4



APPENDIX G
SUBROUTINE CHEAD OUTPUT

```
ID=OS
TYPE   1
CSAD   FOUND
FORMAT=CSAD   STORED IN COMMON   NTEM=    6
CTIB   FOUND
FORMAT=CTIB   STORED IN COMMON   NTEM=    3
CIDB   FOUND
FORMAT=CIDB   STORED IN COMMON   NTEM=    5
CTDB   FOUND
FORMAT=CTDB   STORED IN COMMON   NTEM=    4
RDRD   FOUND
FORMAT=RDRD   STORED IN COMMON   NTEM=   18
RDRM   FOUND
FORMAT=RDRM   STORED IN COMMON   NTEM=  127
RDRT   FOUND
FORMAT=RDRT   STORED IN COMMON   NTEM=   10
FMSP   FOUND
PAC2   FOUND
APC3   FOUND
APC6   FOUND
PA05   FOUND
PA63   FOUND
TRHD   FOUND
TRMA   FOUND
TRTG   FOUND
TRMI   FOUND
TRSP   FOUND
RDID   FOUND
FORMAT=RDID   STORED IN COMMON   NTEM=    4
AACC   FOUND
FORMAT=AACC   STORED IN COMMON   NTEM=    8
AMP1   FOUND
AMP2   FOUND
AMP3   FOUND
AMP4   FOUND
AMP5   FOUND
AMP6   FOUND
APC1   FOUND
ASLP   FOUND
ASMP   FOUND
ATRK   FOUND
AXSP   FOUND
FORMAT=AXSP   STORED IN COMMON   NTEM=   29
BCAL   FOUND
BIAS   FOUND
FORMAT=BIAS   STORED IN COMMON   NTEM=   19
BSMC   FOUND
CATF   FOUND
FORMAT=CATF   STORED IN COMMON   NTEM=   23
CHAF   FOUND
CMDB   FOUND
FORMAT=CMDB   STORED IN COMMON   NTEM=   51
GLOT   FOUND
FORMAT=GLOT   STORED IN COMMON   NTEM=    7
NOM1   FOUND
NOM2   FOUND
NOM3   FOUND
NOM4   FOUND
OBJT   FOUND
PHA1   FOUND
PHA2   FOUND
PHA3   FOUND
PHA4   FOUND
PHA5   FOUND
PHA6   FOUND
RCHF   FOUND
FORMAT=RCHF   STORED IN COMMON   NTEM=    6
R4CH   FOUND
R5CH   FOUND
FORMAT=R5CH   STORED IN COMMON   NTEM=   32
RR04   FOUND
RR05   FOUND
FORMAT=RR05   STORED IN COMMON   NTEM=    8
RR06   FOUND
RR07   FOUND
RR08   FOUND
RR09   FOUND
ERRO   FOUND
SCAN   FOUND
XSEC   FOUND
DRG1   FOUND
DRG2   FOUND
DRG3   FOUND
SASC   FOUND
RR11 NOT FOUND
CHEAD COMPLETE
```

APPENDIX H
SUBROUTINE LTIME PROGRAM LISTING

```
SUBROUTINE LTIME(LOT,IH,IM,IS,IT)
DIMENSION LOT(3),ICON(5)
DATA ICON/Z10,Z100,Z1000,Z10000,Z100000/
IS1 = MOD(LOT(2),ICON(1))
IS10 = MOD(LOT(2)/ICON(1),ICON(1))
IS = IS10*10 + IS1
IM1 = MOD(LOT(2)/ICON(2),ICON(1))
IM10 = MOD(LOT(2)/ICON(3),ICON(1))
IM = IM10*10 + IM1
IH1 = MOD(LOT(2)/ICON(4),ICON(1))
IH10 = MOD(LOT(2)/ICON(5),ICON(1))
IH = IH10 * 10 + IH1
ITU = MOD(LOT(3),ICON(1))
ITT = MOD(LOT(3)/ICON(1),ICON(1))
ITH = MOD(LOT(3)/ICON(2),ICON(1))
IT = ITU + ITT*10 + ITH*100
RETURN
END
```

APPENDIX J
SUBROUTINE GMTTAL PROGRAM LISTING

```
      SUBROUTINE GMTTAL (IGMTH. IGMTM, GMTS, TAL)
      COMMCN /LAUNCH/ TLCNCH
      INTEGER * 2 IGMTH, IGMTM
      REAL * 8 GMTS, TAL
      REAL * 8 TLONCH
      DATA I2T20 /1048576/, I2T16 /65536/, I2T12 /4096/, I?T8 /256/,
     1I2T4 /16/
C
C COMPUTE SECONDS AFTER LAUNCH
C
 100  TAL = IGMTH * 3600 + IGMTM * 60 + GMTS - TLCNCH
C
C CHECK TC SEE IF TAL IS NEGATIVE. IF SO, CHECK FCR MIDNITE LAUNCH
C
      IF (TAL.LT.0.) GC TC 10
      RETURN
  10  IF(DABS(TAL).GE.837C0.0) GO TO 11
      TAL = IGMTH * 36C0 + IGMTM * 60 + GMTS - TLCNCH
      RETURN
  11  TAL=TAL+86400.
      RETURN
C
C
      ENTRY CATIME (FMCATF, IADD, IGMTH, IGMTM, GMTS, TAL)
C
      J = IGET(FMCATF, IADD, 3)
      IGMTH = (J / I?T20) * 10 + (J - (J / I2T20) * I2T20) / I2T16
      IGMTM = ((J - (J / I2T16) * I2T16) / I2T12) * 10 +
     1(J - (J / I2T1?) * I2T12) / I2T8
      GMTS = ((J -(J / I2T8) * I2T8) / I2T4) * 10 +
     1(J - (J / I2T4) * I2T4)
      J = IGET(FMCATF, IADD, 4)
      GMTS = GMTS + (J / I2T8) * .1 + ((J - (J / I2T8) * I2T8) / I2T4)
     1* .01 + (J - (J / I2T4) * I2T4) * .001
C
C CONVERT
C
      GO TO 10C
C
C
      ENTRY RADART (FMRDRM, IRDRM, IGMTH, IGMTM, GMTS, TAL)
C
      J = IGET(FMRDRM, IRDRM, 1)
      IGMTH = (J / I?T12) * 10 + (J - (J / I2T12) * I2T12) / I2T8
      IGMTM = ((J - (J / I2T8) * I2T8) / I2T4) * 10 +
     1(J - (J / I2T4) * I2T4)
      J = IGET(FMRDRM, IRDRM, 2)
      GMTS = (J / I2T16) * 1C + (J - (J / I2T16) * I2T16) / I2T12 +
     1((J - (J / I2T12) * I2T12) / I2T8) * .1 +
     2((J - (J / I2TR) * I2T8) / I2T4) * .01 +
     3(J - (J / I2T4) * I2T4) * .001
C
C CONVERT
C
      GO TO 100
      END
```

APPENDIX K
SUBROUTINE REFC PROGRAM LISTING

VERSION: 6/16/70

```
       SUBROUTINE REFC(E,R,DEE,ORR)
       DIMENSION OE(16,8),OR(16,8),EO(16),RD(8)
       OATA DE/0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,
      10.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0    ,0.0313,
      20.0303,0.0292,0.0287,0.0282,0.0272,0.0262,0.0253,0.0243,0.0223,
      30.0214,0.0195,0.0171,0.0135,0.0075,0.0    ,0.0937,0.0848,0.0770,
      40.0732,0.0694,0.0627,0.0571,0.0522,0.0480,0.0412,0.0385,0.0337,
      50.0278,0.0205,0.0105,0.0    ,0.1850,0.1520,0.1250,0.1140,0.1050,
      60.0904,0.0795,0.0708,0.0636,0.0523,0.0478,0.0405,0.0323,0.0229,
      70.0114,0.0    ,0.5310,0.3070,0.2120,0.1830,0.1600,0.1280,0.1060,
      80.0899,0.0780,0.0612,0.0550,0.0455,0.0354,0.0246,0.0120,0.0    ,
      90.7550,0.3720,0.2400,0.2020,0.1750,0.1370,0.1120,0.0942,0.0811,
      A0.0631,0.0566,0.0466,0.0361,0.0250,0.0122,0.0    ,0.9120,0.4110,
      B0.2560,0.2140,0.1840,0.1420,0.1150,0.0967,0.0830,0.0643,0.0575,
      C0.0472,0.0365,0.0252,0.0122,0.0    ,0.9700,0.4200,0.2600,0.2200,
      D0.1900,0.1460,0.1170,0.0980,0.0840,0.0653,0.0584,0.0478,0.0369,
      E0.0254,0.0123,0.0 /
       OATA DR/  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,
      1  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  C.0, 22.6, 21.5, 20.4, 19.9,
      2 19.4, 18.5, 17.6, 16.8, 16.1, 14.8, 14.2, 13.2, 12.0, 10.4,  8.6,
      3  7.7, 67.3, 57.9, 50.2, 47.0, 44.1, 39.3, 35.4, 32.1, 29.3, 24.8,
      4 22.9, 19.7, 16.3, 12.7,  9.4,  8.1,132.0, 98.5, 77.4, 69.7, 63.2,
      5 52.9, 44.7, 38.4, 33.4, 26.4, 23.9, 20.1, 16.4, 12.7,  9.4,  8.1,
      6340.0,167.0,103.0, 86.1, 73.4, 56.7, 46.2, 38.9, 33.6, 26.4, 24.0,
      7 20.2, 16.4, 12.8,  9.5,  8.2,405.0,170.0,104.0, 86.3, 73.6, 56.8,
      8 46.3, 38.9, 33.7, 26.5, 24.1, 20.3, 16.5, 12.8,  9.5,  8.2,421.0,
      9171.0,104.0, 86.6, 73.9, 57.1, 46.4, 39.0, 33.8, 26.8, 24.3, 20.5,
      A 16.6, 13.0,  9.8,  8.4,446.0,172.0,105.0, 87.4, 74.0, 58.0, 46.6,
      8 39.2, 34.0, 27.0, 24.6, 20.7, 16.7, 13.0, 10.0,  8.4/
       DATA EO,RTOEG/0.01,2.0,4.0,5.0,6.0,8.0,10.0,12.0,14.0,18.,20.,
      124.,30.,40.,60.,90.,57.29578/
       DATA RO/0.01,10.,30.,60.,200.,400.,1000.,2000./
       IF(R.LE.0.C)GO TC 300
       RG=R/6080.27
       OO 100 IEO=2,15
       I=17-IEO
       IF(E.GE.EO(I))GO TO 120
100    CCNTINUE
       I=1
120    OO 200 JRO=2,8
       J=10-JRD
       IF(RG.CE.RO(J))GC TO 220
200    CCNTINUE
       J=1
220    IF(J.EC.8)GO TO 34C
       ZR=ALOG(RG/RO(J))/ALOG(RD(J+1)/RD(J))
       IF(E.LE.0.0)GO TC 320
       ZE=ALOG(E/ED(I))/ALOG(ED(I+1)/ED(I))
       OE1=((DE(I+1 J)-DE(I,J))*(1.-ZR)+(OE(I,J+1)-DE(I,J))*ZR)*ZE
       OE2=((OE(I,J+1)-OE(I,J))*(1.-ZE)+(DE(I+1,J+1)-DE(I,J+1))*ZE)*ZR
       OEE=OE1+OE2+DE(I,J)
       OR1=((DR(I+1,J)-CR(I,J))*(1.-ZR)+(DR(I,J+1)-DR(I,J))*ZR)*ZE
       OR2=((DR(I,J+1)-CR(I,J))*(1.-ZE)+(DR(I+1,J+1)-DR(I,J+1))*ZE)*ZR
       ORR=(DR1+DR2+OR(I,J))
       GO TO 400
300    DEE=0.0
       ORR=0.0
       GO TO 400
320    DEE=OE(I,J)+(OE(I,J+1)-OE(I,J))*ZR
       ORR=DR(I,J)+(OR(I,J+1)-DR(I,J))*ZR
       GO TO 400
340    OELT=(E-ED(I))/(ED(I+1)-ED(I))
       OEE=DELT*(DE(I+1,J)-DE(I,J))+DE(I,J)
       DRR=OELT*(DR(I+1,J)-DR(I,J))+OR(I,J)
400    RETURN
       ENO
```

APPENDIX L
SUBROUTINE BZERO PROGRAM LISTING

```
                START  0
                ENTRY  BZERO
                USING  *,15
BZERO           B      H5
                DC     X'05',CL5'BZERO'
H5              STM    14,12,12(13)
                L      6,0(1)
                MVC    WORD,0(6)
                LE     4,WORD
                LPER   6,4
                CE     6,=E'1.'
                BL     H20
                AU     4,X6
                STE    4,WORD
                L      7,WORD
                SR     9,9
                LTR    7,7
                SLL    7,8
                BNL    H10
                LA     9,1
H10             SRL    7,8
                SR     2,2
                AH     2,=X'0103'
                SRDL   2,2
                SRL    3,30
                SLL    7,0(3)
                SRL    7,2
                SRDL   2,8
                OR     7,3
                SLL    9,31
                OR     7,9
                SER    2,2
                ST     7,WORD
                AE     2,WORD
                STE    2,WORD
                MVC    0(4,6),WORD
H20             RETURN (14,12)
WORD            DS     1E
X6              DC     X'46000000'
                END
```

APPENDIX M
FUNCTION IBIT PROGRAM LISTING

```
      FUNCTION IBIT (ITEM, NBIT)
      INTEGER * 2 ITFM
      LIMIT = NBIT - 2
      J = 0
100   IF (ITEM .EQ. (2**J) ) GO TO 120
      IF (J .GE. LIMIT) GO TO 110
      J = J + 1
      GO TO 100
110   IBIT = NBIT
      RETURN
120   IBIT = J + 1
      RETURN
      END
```